Spektrum Industri
Vol. 22, No. 2, 2024, pp. 163-178
ISSN 1693-6590
http://journal3.uad.ac.id/index.php/spektrum

Facility Layout Planning of Sheet Metal Working Industry Using Metaheuristics

Adinda Sekar Ludwika [a], Maratus Shalehah [a], Rakan Raihan Ali Mohammad [a], Andiny Trie Oktavia [a], Nur Mayke Eka Normasari [a], Nguyen Huu Tho [b], Achmad Pratama Rifai [a,*]

[a] Department of Mechanical and Industrial Engineering, Universitas Gadjah Mada, Jl. Grafika No. 2, Yogyakarta 55281, Indonesia
[b] Faculty of Engineering and Technology, Nguyen Tat Thanh University, Ho Chi Minh City, Vietnam
* Corresponding Author: achmad.p.rifai@ugm.ac.id

ARTICLE INFO

Article history
Received July 21, 2023
Revised October 1, 2024
Accepted October 23, 2024

Keywords
Facility layout;
Simulated Annealing;
Large neighborhood search;
Ant Colony Optimization

ABSTRACT

The design of a facility layout on the production floor is critical for ensuring the efficiency and effectiveness of production processes. Poorly planned layouts can disrupt production flow, increase operational costs, and negatively impact productivity. In job shop production environments, where diverse products with varying process flows are manufactured, optimizing the layout becomes even more essential. This research addresses the single row facility layout problem in a sheet metal working industry, focusing on minimizing the total material handling distance. Metaheuristic algorithms, including Simulated Annealing (SA), Large Neighborhood Search (LNS), Adaptive Large Neighborhood Search (ALNS), and Ant Colony Optimization (ACO), were employed to achieve optimal layout configurations. The SA, LNS, and ALNS algorithms yielded the best results, with a total material handling distance of 897,171 meters and an optimal facility arrangement of either 7-5-6-4-3-2-1 or 1-2-3-4-6-5-7. Among these, SA proved to be the most efficient in terms of computational time, making it the preferred algorithm for solving this layout problem.

1. Introduction

Facility layout is a well-known challenge in industrial engineering, as an effective layout can significantly improve a company's performance (Alberto & Geoff, 1998). A well-organized facility layout enhances operational efficiency and can reduce total production costs by as much as 50% (Tompkins et al., 1996). Facility layout involves arranging all the essential components needed for the production of goods or services. These components include machines, workstations, divisions, warehouses, and other resources that support the execution of work tasks (Heragu & Kakuturi, 1997).

The primary goal of layout planning is to optimize workflow and minimize costs, particularly those associated with material handling and the psychological impact on workers. This is especially important for manufacturing companies with a job shop production flow, which are characterized by a high variety of product types. For instance, a sheet metal working company may produce up to 50 different product types, each with its own unique fabrication process flow, within a single week. This variation often leads to inefficiencies, such as overlapping processes and a decline in operator performance, as frequent material movements can disrupt the sequence of operations and hinder productivity.

There are various methods for solving facility layout problems, ranging from conceptual methods (Automated Layout Design Program, systematic layout planning) to heuristic methods (group technology, cell manufacturing, SLP, genetic algorithms, electre method, simulated annealing, etc.) and exact programming methods (ABS, MILP, NLP, etc.). Among these, conceptual approaches like the Automated Layout Design Program (ALDEP) and systematic layout planning (SLP) have been widely applied in industries where the closeness of relationships between departments is the primary concern. While these methods are simple and effective for small-scale layouts, they often lack the robustness needed for complex, larger-scale problems or situations where minimizing the total travel distance is critical. For instance, ALDEP has been successfully applied in pyrolyzer production layout design (Rifai et al., 2023), but it is limited by its heuristic nature, which may not always yield optimal solutions for more intricate problems.

On the other hand, metaheuristic methods such as Simulated Annealing (SA), Genetic Algorithms (GA), and Large Neighborhood Search (LNS) offer stronger optimization capabilities, especially in dealing with more complex layouts. Simulated Annealing, for example, has been widely used in various industries to solve combinatorial optimization problems such as job shop scheduling and layout planning. Its strength lies in its ability to escape local optima by accepting worse solutions at a controlled probability, making it highly effective in searching large solution spaces (Dehghan-Sanej et al., 2021). Agista et al. (2021) employed Simulated Annealing along with Dimensionless Block Diagram and Modified Spanning Tree for layout planning in woodcraft industry. The methods aim to minimize the total travelled distance of material handling. Kusumaningsih et al. (2022) also used Simulated Annealing for layout planning in furniture industries. Two layout approaches are explored, which are single and double row layout. However, SA's reliance on carefully tuned parameters and longer computation times can be a drawback in scenarios requiring fast decision-making.

Although the single row layout is the most common type of production layout in industries, especially in small and medium enterprises, the double row layout is proven to provide better access for material handling. Rifai et al. (2020) proposed a genetic algorithm for the double row layout problem with the objective of minimizing total travelled distance. Genetic Algorithms (GA) are another popular choice for solving layout problems due to their ability to explore diverse solutions through genetic operators like mutation and crossover. Although GA can produce highly optimized solutions, its performance is sensitive to the chosen population size and convergence criteria, which can lead to either excessive computation times or suboptimal solutions if not properly configured (Rifai et al., 2020). Moreover, GAs may require significant computational resources when dealing with large-scale, high-dimensional problems.

More recently, hybrid metaheuristics emerge as alternative approaches for layout planning. In a recent study, Rifai et al. (2022) proposed a two-stage variable neighborhood search approach for the double row layout problem with safety consideration. The proposed two stage method was based on the improved variable neighborhood search (IVNS) and sine-cosine algorithm (SCA), both are metaheuristics. Other metaheuristic methods that have been employed for solving the facility layout problem are migrating bird optimization and tabu search (Tongur et al., 2020), A* search algorithm (Besbes et al., 2021), constrained memetic algorithm (Liu et al., 2021), and Cuckoo search algorithm (Maghfiroh et al., 2023). Rifai et al. (2023) developed SA and Modified Spanning Tree (MST) for solving the single row and double row layout problems in bakery industry. In a recent study, Maier and Taverner (2023) proposed a novel Integer Linear Programming (ILP) formulation for SRLP by considering positioning, ordering, and relation constraints on single-row facility layouts. The study indicated that although SRLP is one of the most studied facility layout problems in the literature, it is still an active domain in which novel methods, either conceptual methods, exact programming, or heuristics are still developed.

While exact programming methods such as Mixed-Integer Linear Programming (MILP) can provide optimal solutions, they are generally impractical for large-scale problems due to their high computational costs and exponential growth in solution time as the problem size increases. Isnaini et al. (2024) evaluated BLOCPLAN, exact algorithm, and Particle Swarm Optimization (PSO) for solving the double row layout problem in manufacturing industry. The results indicated that PSO performed the best in complex scenarios with higher numbers of machines. Therefore, for many real-world applications, the trade-off between solution quality and computational feasibility makes metaheuristics the preferred choice.

However, despite their advantages, there are still ongoing challenges in implementing metaheuristics in industrial settings. One key issue is the need for computational efficiency, as many metaheuristic methods still require considerable processing time, especially when dealing with real-time layout adjustments or highly dynamic production environments. Additionally, balancing the exploration of new solutions and exploiting known good solutions remains a critical challenge, as many methods may either converge too quickly or fail to find global optima.

While metaheuristics are highly effective for addressing the facility layout problem, ongoing research must focus on improving their computational efficiency. As such, this study contributes to the field by applying and critically evaluating several metaheuristic algorithms for solving the single row layout problem (SRLP), which are Simulated Annealing (SA), Large Neighborhood Search (LNS), Adaptive Large Neighborhood Search (ALNS), and Ant Colony Optimization (ACO). ALNS, the improved version of LNS, is particularly effective for problems that require balancing between exploration and exploitation, as it can adaptively switch between different operators depending on their past effectiveness. Meanwhile, ACO has been previously applied in various layout problems with promising results. Each of these methods is assessed in terms of their performance, computational efficiency, and ability to produce optimal layout configurations.

2. Method

This study aims to determine the most effective production layout in a company using metaheuristic methods with several algorithms. The algorithms used are SA, LNS, ALNS, and ACO. In this research, four different algorithms are employed to compare their performance in generating the best production layout in terms of arrangement and total distance. The algorithms are chosen because they are designed to solve optimization problems in discrete forms, which align with the discrete variables involved in the facility layout problem in this case study. The algorithms are developed and written in MATLAB.

2.1. Research framework

The research framework of this study is presented in Fig. 1. The study begins by clearly defining the Single Row Facility Layout Problem, where the primary objective is to minimize the total material handling distance between various machines and production areas. This stage also involves outlining the constraints, such as machine sizes, facility distances, and material flow, which are essential for optimizing the layout. To accomplish this, data is first collected, including the dimensions of the facilities involved and the material flow matrix, which indicates the frequency of material movement between different facilities.

Following the data collection, the next crucial step is the development of the mathematical model. The model is formulated with the objective of minimizing the total material handling distance, expressed through an equation that sums the distances between facility pairs weighted by the material flow between them. Constraints are also defined to ensure that facilities are arranged appropriately without overlapping and that distances between them are correctly calculated based on their lengths and relative positions.

Once the problem is mathematically defined, an initial layout is created to serve as a baseline for optimization. This initial layout is a basic configuration of the facilities that the optimization algorithms will refine through successive iterations. The core of the optimization process involves applying various metaheuristic algorithms, specifically SA, LNS, ALNS, and ACO. These algorithms are chosen for their effectiveness in solving complex combinatorial optimization problems like the SRFLP. Each algorithm is designed to iteratively adjust the facility layout, aiming to reduce the material handling distance by optimizing the layout configuration.

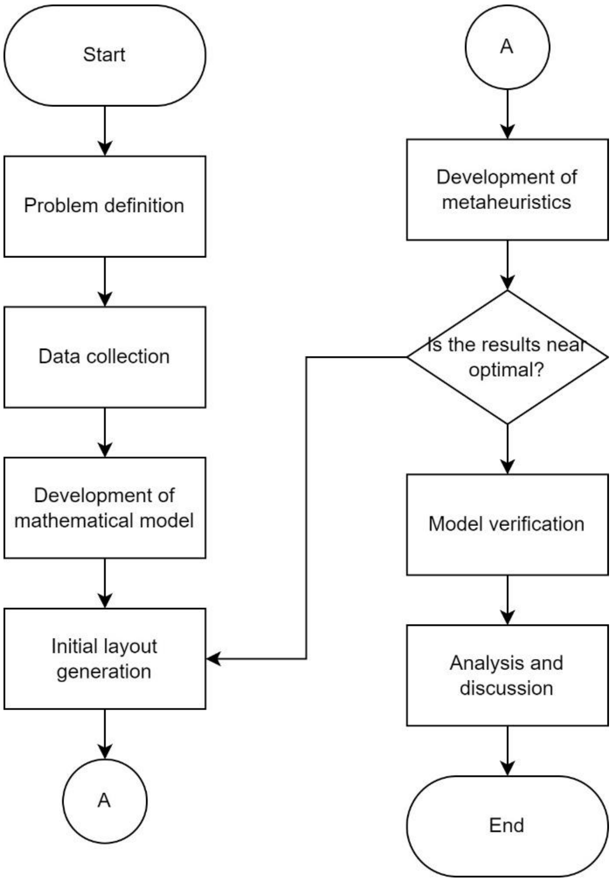

Fig. 1. Research framework

The algorithms are then evaluated based on their performance, including the total material handling distance they achieve, their computational efficiency, and the time taken to converge to a solution. SA, LNS, and ALNS tend to produce more optimal results with shorter material handling distances, while ACO, though useful in some contexts, often results in higher total distances due to its heavier focus on exploration rather than exploitation.

Afterward, a model verification process is conducted to ensure the accuracy of the results generated by the algorithms. The outcomes are cross-checked against manual calculations to verify that the total material handling distances are correct, ensuring that the algorithms function as expected.

The final stage of the method is the Analysis and Discussion. Here, the results from each algorithm are critically analyzed. The superiority of SA, LNS, and ALNS over ACO is emphasized, particularly in terms of producing layouts with significantly shorter material handling distances. The discussion explores the reasons for ACO’s less optimal performance, noting that its tendency to prioritize exploration over exploitation led to slower convergence and a less efficient solution. The strengths and weaknesses of each algorithm are discussed, with SA emerging as the most computationally efficient and effective in balancing exploration and exploitation.

2.2. Mathematical Model

The given mathematical model represents a formulation for the Single Row Layout Problem formulated by Amaral (2006), aiming to minimize the total weighted distance between facilities placed on a single row.

Objective function:

$$\sum_{i=1}^{M-1} \sum_{j=i+1}^{M} d_{ij} F_{ij} \tag{1}$$

Subject to:

$$\sum_{k<i} l_k \alpha_{ki} + \sum_{k>i} l_k (1 - \alpha_{ik}) - \sum_{k<j} l_k \alpha_{kj} - \sum_{k>i} l_k (1 - \alpha_{jk}) \ + (l_i - l_j)/2 \\ \forall i, j, k \in N, 1 \leq j < j \leq M \tag{2}$$

$$\sum_{k<i} l_k \alpha_{ki} - \sum_{k>i} l_k (1 - \alpha_{ik}) + \sum_{k<j} l_k \alpha_{kj} + \sum_{k>i} l_k (1 - \alpha_{jk}) \ + (l_j - l_i)/2 \\ \forall i, j, k \in N, 1 \leq j < j \leq M \tag{3}$$

$$\alpha \in H_M \tag{4}$$

$$d \in D_M \tag{5}$$

$$\alpha_{ij} \in \{0,1) \\ \forall i, j \in N, 1 \leq j < j \leq M \tag{6}$$

Where M is number of facilities, N is the set of facilities, d_{ij} is the distance between department i and j, and F_{ij} is the material handling frequency from department i to j. The objective is to minimize the total weighted sum of distances between all pairs of facilities i to j, as formulated in Eq. (1).

The length of facility k is denoted by l_k, while α_{ij} is an additional binary variable where it has the value of 1 if facility i is placed before facility j, 0 otherwise. Eq. (2) to Eq. (3) ensure that the distance d_{ij} between facilities i and j is bounded by the facility lengths and their relative positioning. These constraints ensure that the facilities are arranged in such a way that their lengths and the distances between them do not allow overlap. Eq. (4) indicates that the decision variables α belong to a feasible set of layout decisions. Eq. (5) denotes that the distances d_{ij} are valid, non-negative distances based on the layout configuration. At last, Eq. (5) defines the binary decision variable α_{ij}.

2.3. Simulated Annealing

Simulated Annealing has been widely applied to solve optimization problems, such as in the job shop scheduling (Dehghan-Sanej et al., 2021) and travelling salesman problem (Nugracia & Muslim Lhaksmana, 2020). Simulated Annealing was chosen for its robust capability in escaping local optima by accepting worse solutions with a certain probability, a feature particularly useful for complex facility layout problems where the solution space is large and contains many local minima. SA is also known for its relatively simple implementation and adaptability to various types of optimization problems. One of its key advantages is its ability to balance exploration and exploitation, ensuring thorough examination of the solution space before converging on an optimal solution. Although SA may require careful parameter tuning and can be slower to converge, it has consistently demonstrated strong performance in layout optimization problems, especially when computational speed is less critical than solution quality.

The general steps of the SA algorithm are illustrated in Fig. 2. The pseudocode of the SA algorithm provides a step-by-step guide for this research, as follows:

1. Input SA parameters: The parameters used in this research are the initial temperature (T0) set to 200, the final temperature (TF) set to 1, and the cooling factor (α) set to 0.9738.
2. Create an initial solution.
3. Calculate the fitness value for the current layout.
4. Begin the SA iteration process.
5. If the temperature is still higher than the final temperature, modify the solution, calculate the new fitness value, calculate the difference in fitness value between the final and initial solutions, and update the temperature using the equation $T = \alpha T$.
6. Iteration is completed, and the best new layout along with its fitness value is obtained.

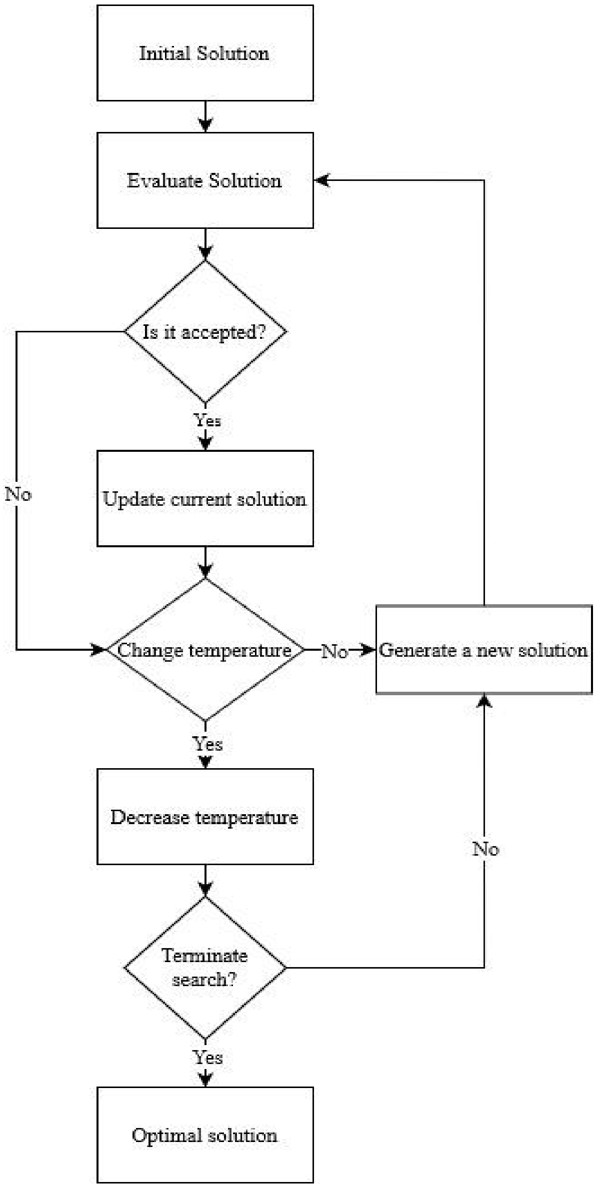

Fig. 2. Flowchart of Simulated Annealing

2.4. Large Neighborhood Search

The LNS algorithm is a search algorithm that utilizes local search methods to find the best solution (Pisinger & Ropke, 2019). It searches for solutions based on the neighborhood of the initial solution. In the LNS algorithm, the evaluation process involves creating the neighborhood of an initial solution, selecting a solution from that neighborhood to evaluate based on the objective function, and comparing it with the initial solution until the best solution is reached.

LNS was selected for its effectiveness in exploring large portions of the solution space through its destruction and repair mechanisms. This algorithm is particularly useful in handling facility layout problems with many discrete variables, as it systematically searches for better solutions by making significant alterations to the current solution before gradually refining it. Compared to other metaheuristics like GA or PSO, LNS excels in problems where larger changes to the solution are necessary to escape local optima. The algorithm's flexibility in handling complex, multi-dimensional layout configurations made it a fitting choice for this study.

The pseudocode of the LNS algorithm provides a step-by-step guide for this research, as follows:

1. Input parameters and data: The parameters used in this research are the number of iterations (T) set to 200, the destruction degree (D) set to 0.1 and 0.5, the number of departments (Dept) set to 7, the width of each department (Dept_width) set to [1.5, 2, 1, 1.5, 3, 4, 4], and the clearance between departments (clearance) set to 2.
2. Create an initial solution.
3. Calculate the fitness value of the initial solution.
4. Perform iterations, which consist of the destruction and repair processes.
5. Calculate the fitness value of the updated solution.
6. Create an acceptance criterion to determine if the updated solution is better than the previous solution.

2.5. Adaptive Large Neighborhood Search

The ALNS is an extension of LNS, with adaptive weight calculation for destruction and repair. ALNS is included for its adaptive capabilities that allow it to dynamically adjust its search parameters based on the performance of previous iterations. This feature provides a more balanced search strategy between exploration and exploitation, which is particularly valuable for problems with dynamic or evolving constraints, such as the facility layout problem. ALNS has been shown to outperform traditional LNS in terms of convergence speed and solution quality, making it a more efficient option for large-scale industrial optimization problems. Its adaptability and superior performance in practical applications justified its inclusion in this study over other methods such as PSO, which may struggle with balancing exploration and exploitation in complex solution landscapes.

There are 2 destroy and 3 repair operators used. The alpha parameter is set to 0.9, while beta parameter is set to [1.3, 0.8]. Unlike the LNS in which the operators of destroy and repair for solution modifications are determined randomly, in the ALNS, the operators are selected based on their historical performance in previous iterations. As such, the operators which have better historical performance will have higher chance to be selected.

2.6. Ant Colony Optimization

The Ant Colony Optimization is a popular metaheuristic that has been applied to solve a wide array of optimization problems, such as in travelling salesman problem (Udjulawa et al., 2022) and assignment problem (Gandhi & Widyawati, 2019). ACO is chosen for its strong track record in solving combinatorial optimization problems, such as the traveling salesman problem, which shares similarities with the facility layout problem. ACO mimics the behavior of ants searching for optimal paths, utilizing pheromone trails to guide the search process. This makes ACO particularly wellsuited for problems that require finding optimal paths or sequences, such as determining the most efficient layout of facilities. Its ability to continuously improve solutions based on pheromone updates allows for effective exploration of the solution space. ACO, however, can be slower in terms of computational time compared to SA or LNS, but its strength lies in its ability to find high-quality solutions in complex, multi-variable problems. Given the discrete nature of the facility layout problem and ACO's proven effectiveness in similar optimization tasks, it was considered an appropriate choice for this study.

In this study, the ACO algorithm implemented in MATLAB follows the following steps:

1. Set the initial values for parameters in the ACO algorithm, such as the number of ants, maximum iterations, alpha, beta, evaporation rate (rho), and pheromone deposit factor (Q).
2. Initialize the pheromone values.
3. Calculate the heuristic values.
4. ACO Iteration:
    1. Initialize ant colonies by creating a population of ants with random positions along possible solutions.
    2. Ant movement: Each ant selects the next step based on probabilistic rules using the heuristic values and pheromone concentrations on available paths.
    3. Calculate the total material handling distance between facilities based on the formed ant paths.
    4. Evaluate the ant paths based on the given objective function, which is the shortest total material handling distance for all facilities.
    5. Update the pheromone values, updating the pheromone level on each path based on the evaluation of ant paths. The better the path taken by ants, the greater the increase in pheromone on that path.
5. If the termination criteria, such as reaching the maximum number of iterations, are met, stop the computation, and display the best solution.

3. Results and Discussion

3.1. Dataset

The production floor in the sheet metal working company consists of 7 types of facilities, with 4 of them being machines and the other 3 being production areas that use small machines or involve manual operations (Retnowati & Fudhla, 2013). Table 1 provides an overview of the types of facilities and their dimensions.

Table 1. Types of Facilities

Number	Facilities Type	Dimension (meter)	
		Length	Width
1	Shearing machine	3	1.5
2	Punching 1 machine	1.5	2
3	Punching 2 machine	0.5	1
4	Bending machine	3	1.5
5	Grinding Area	3	3
6	Welding Area	4	4
7	Assembly Area	4	4

The manufacturing company engaged in sheet metal working operates with a job shop production flow, where it produces a variety of 50 different products in a week, each with a distinct fabrication process flow. Table 2 describes the flow matrix between facilities over a span of one year, from July 2011 to July 2012.

Table 2. From-to chart

Facilities	1	2	3	4	5	6	7
1	0	6.268	6.331	4.431	121	598	0
2	6.268	0	4.905	7.189	2.084	4.129	233
3	6.331	4.905	0	6.656	1.719	3.777	423
4	4.431	7.189	6.656	0	1.129	5.556	440
5	121	2.084	1.719	1.129	0	3.563	3.563
6	598	4.129	3.777	5.556	3.563	0	0
7	0	233	423	440	3.563	0	0

3.2. Result of Simulated Annealing

Based on the results obtained from the Simulated Annealing algorithm using MATLAB software for production layout, the optimal production layout sequences are machine 1-2-3-4-6-5-7 and 7-5-6-4-3-2-1, with a total distance of 897.171 meters. Table 3 shows the results of SA in ten replications, while Fig. 3 presents the optimization process using SA.

Table 3. Results of SA Method Replication

Replication	Department order	Total distance (Meter)	Computation time (second)
1	7 5 6 4 3 2 1	897.171	0.0938
2	7 5 6 4 3 2 1	897.171	0.0001
3	1 2 3 4 6 5 7	897.171	0.0469
4	1 2 3 4 6 5 7	897.171	0.0781
5	1 2 3 4 6 5 7	897.171	0.0312
6	1 2 3 4 6 5 7	897.171	0.0625
7	7 5 6 4 3 2 1	897.171	0.0001
8	7 5 6 4 3 2 1	897.171	0.0001
9	7 5 6 4 3 2 1	897.171	0.0469
10	7 5 6 4 3 2 1	897.171	0.0001
Average			0.0250

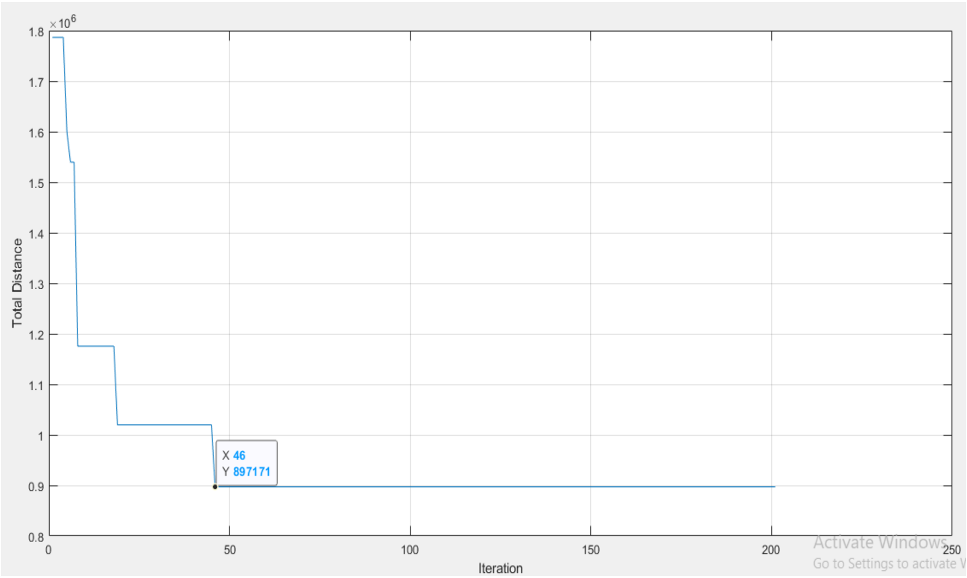

Fig. 3. Simulated Annealing Chart

Fig. 3 presents an example of a graph showing the results from one of the replications conducted. The graph indicates that after 200 iterations, it is observed that starting from iteration 46, the results stabilize with no further changes in the total distance value. Therefore, it can be concluded that from iteration 46 onwards, the iteration process of SA has reached convergence.

3.3. Result of Large Neighborhood Search

After conducting the layout search using the Large Neighborhood Search algorithm using MATLAB software for production layout, different results were obtained in terms of the production layout sequence and the total distance of the layout achieved. Table 4 shows the results of LNS in ten replications, while Fig. 4 presents the optimization process using LNS.

Table 4. Results of LNS Method Replication

Replication	Department order	Total distance (Meter)	Computation time (second)
1	7 5 6 4 3 2 1	897.171	0.5000
2	7 5 6 4 3 2 1	897.171	0.0781
3	7 5 6 4 3 2 1	897.171	0.0625
4	1 2 3 4 6 5 7	897.171	0.0312
5	5 6 2 4 3 1 7	1.026.198	0.0781
6	7 5 6 4 3 2 1	897.171	0.0938
7	7 5 6 4 3 2 1	897.171	0.0938
8	7 5 6 4 3 2 1	897.171	0.0312
9	1 3 4 2 6 5 7	904.314	0.0312
10	7 5 6 4 3 2 1	897.171	0.0312
Average			0.1031

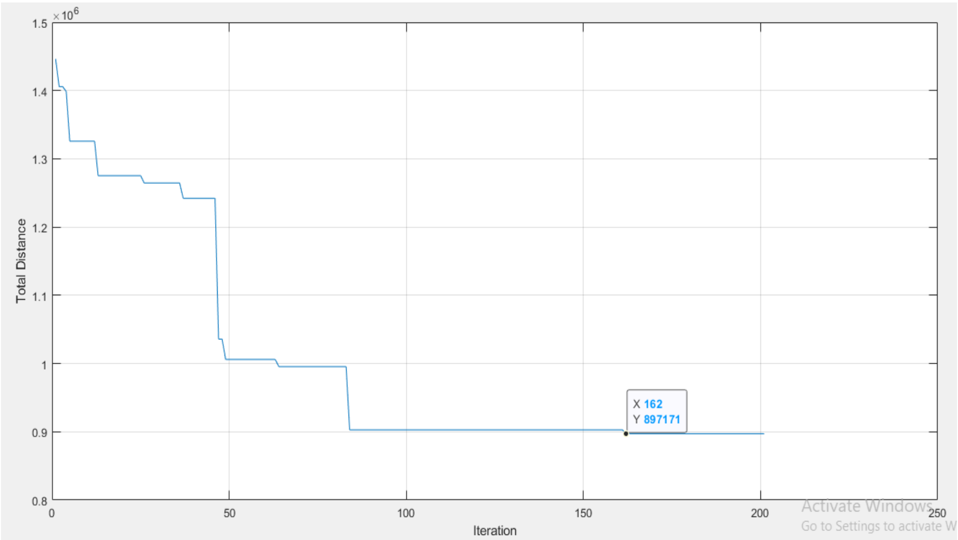

Fig. 4. Large neighborhood search chart

Based on several obtained results, to determine the best outcome, it is done by selecting based on the smallest total distance of the layout. This is done with the aim of selecting the most optimal layout, hence the choice of the smallest distance. Therefore, the best result for determining the layout using the LNS algorithm is the machine sequence of 7-5-6-4-3-2-1 and 1-2-3-4-6-5-7, with a total distance of 897.171 meters. The graph in Fig. 4 shows that after 200 iterations, it is observed that starting from iteration 162, the results stabilize with no further changes in the total distance value. Therefore, it can be concluded that from iteration 162 onwards, the LNS algorithm has reached convergence.

3.4. Result of Adaptive Large Neighborhood Search

After conducting the layout search using the Adaptive Large Neighborhood Search algorithm using MATLAB software for production layout, different results were obtained in terms of the production layout sequence and the total distance of the layout achieved. Table 5 shows the results of ALNS in ten replications, while Fig. 5 presents the optimization process using ALNS.

Table 5. Results of ALNS Method Replication

Replication	Department order	Total distance (Meter)	Computation time (second)
1	7 1 2 3 4 6 5	1.026.805	0.4688
2	5 6 2 3 4 1 7	1.026.198	0.1719
3	7 1 2 3 4 6 5	1.026.805	0.1250
4	5 6 3 4 1 2 7	1.072.366	0.0781
5	1 3 4 2 6 7 5	996.746	0.0938
6	5 6 4 3 2 1 7	1.026.805	0.1406
7	7 5 6 4 3 2 1	897.171	0.0469
8	7 5 6 4 2 3 1	902.921	0.0781
9	1 2 3 4 6 5 7	897.171	0.0625
10	7 1 3 4 2 6 5	1.026.198	0.0625
Average			0.1328

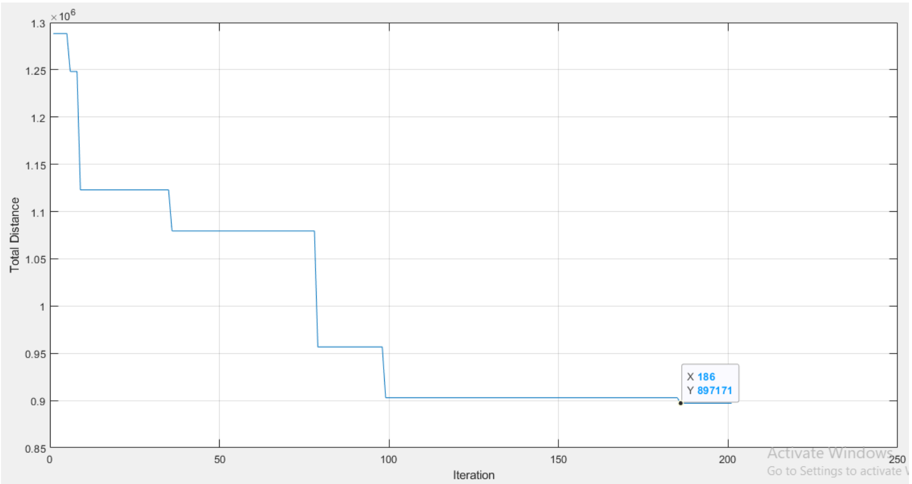

Fig. 5. Adaptive Large Neighborhood Search Chart

Based on several obtained results, to determine the best outcome, it is done by selecting based on the smallest total distance of the layout. This is done with the aim of selecting the most optimal layout, hence the choice of the smallest distance. Therefore, the best result for determining the layout using the ALNS algorithm is the machine sequence of 7-5-6-4-3-2-1 and 1-2-3-4-6-5-7, with a total distance of 897.171 meters. The graph in Fig. 5 shows that after 200 iterations, it is observed that starting from iteration 186, the results stabilize with no further changes in the total distance value. Therefore, it can be concluded that from iteration 186 onwards, the ALNS algorithm has reached convergence.

3.5. Result of Ant Colony Optimization

After conducting the layout search using the Ant Colony Optimization algorithm using MATLAB software for production layout, different results were obtained in terms of the production layout sequence and the total distance of the layout achieved. Table 6 shows the results of ACO in ten replications, while Fig. 6 presents the optimization process using ACO.

Based on several obtained results, to determine the best outcome, it is done by selecting based on the smallest total distance of the layout. This is done with the aim of selecting the most optimal layout, hence the choice of the smallest distance. Therefore, the best result for determining the layout using the ACO algorithm is the machine sequence of 5-7-6-4-3-2-1, with a total distance of 989.603 meters. The graph in Fig. 6 shows that after 200 iterations, it is observed that starting from iteration 11, the results stabilize with no further changes in the total distance value. Therefore, it can be concluded that from iteration 11 onwards, the ACO algorithm has reached convergence.

Table 6. Results of ACO Method Replication

Replication	Department order	Total distance (Meter)	Computation time (second)
1	5 7 6 4 3 2 1	989.603	0.3281
2	7 6 5 4 3 2 1	1.019.891	0.2656
3	7 6 5 4 3 2 1	1.019.891	0.1094
4	7 6 5 4 3 2 1	1.019.891	0.1250
5	7 6 5 4 3 2 1	1.019.891	0.2188
6	7 6 5 4 3 2 1	1.019.891	0.1875
7	5 7 6 4 2 3 1	995.353	0.1250
8	7 6 5 4 3 2 1	1.019.891	0.2344
9	7 6 5 4 3 2 1	1.019.891	0.1250
10	5 7 6 4 3 2 1	989.603	0.2812
Average			0.3890

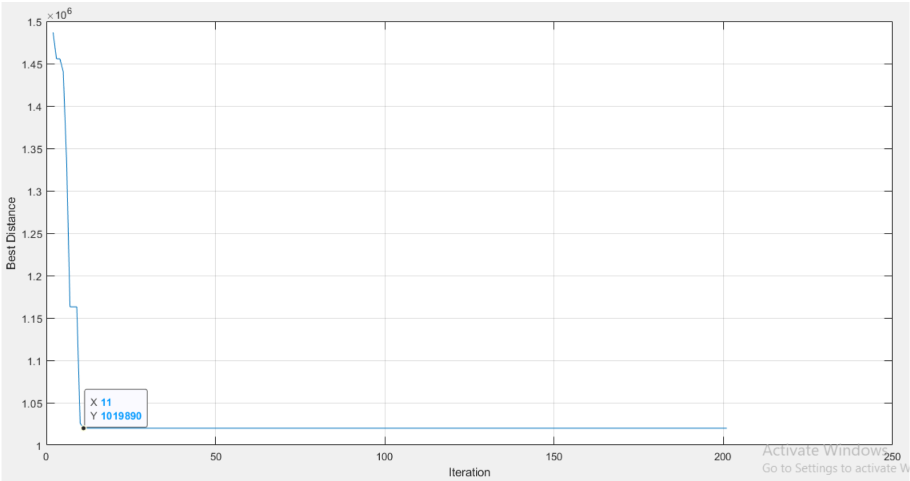

Fig. 6. Ant Colony Optimization chart

3.6. Model Verification

3.6.1. Initial Solution Verification

The models in all four methods (SA, LNS, ALNS, and ACO) share the same coding for the initial solution, so the verification only needs to be conducted once. Verification is done by comparing the results obtained from running the model using the command window in MATLAB with the calculations done using Excel. The variables that will be compared are the travelled distance and the total distance. In this verification process, the initial layout used is the sequence of departments 1-2-3-4-5-6-7. Additionally, a clearance value of 2 meters is utilized. Fig. 7 present an illustration of comparison between the MATLAB and manual calculation results in Excel.

Based on the results from running the model and the calculations in Excel, it was found that the results were approximately the same. The slight differences in values between the two can be attributed to different rounding mechanisms used in MATLAB and Excel. Therefore, the model for calculating the initial solution, including the travelled distance and total distance, can be considered verified.

3.6.2. SA Method Model Verification

Verification was conducted by comparing the results from running the model in MATLAB with the calculations in Excel. The calculated total distance from both MATLAB and Excel was found to be the same, which is 897.171.

```
Command Window
>> travelled_distance

travelled_distance =

   1.0e+04 *

        0    2.3505    4.5900    4.6525    0.1785    1.2109         0
   2.3505         0    1.7167    4.8526    2.2924    6.8129    0.5242
   4.5900    1.7167         0    2.1632    1.2893    4.9101    0.8037
   4.6525    4.8526    2.1632         0    0.4798    5.4171    0.6930
   0.1785    2.2924    1.2893    0.4798         0    1.9596    4.0975
   1.2109    6.8129    4.9101    5.4171    1.9596         0         0
        0    0.5242    0.8037    0.6930    4.0975         0         0

>> Total_distance

Total_distance =

      1019891
```

Travelled distance:	0	23505	45899	46525	1784	12109	0
	23505	0	17167	48525	22924	68128	5242
	45899	17167	0	21632	12892	49101	8037
	46525	48525	21632	0	4798	54171	6930
	1784	22924	12892	4798	0	19596	40974
	12109	68128	49101	54171	19596	0	0
	0	5242	8037	6930	40974	0	0
Total distance:	1019878						

Fig. 7. Comparison of Calculation Results

3.6.3. LNS Method Model Verification

Verification was conducted by comparing the results from running the model in MATLAB with the calculations in Excel. The calculated total distance from both MATLAB and Excel was found to be the same, which is 897.171.

3.6.4. ALNS Method Model Verification

Verification was conducted by comparing the results from running the model in MATLAB with the calculations in Excel. The calculated total distance from both MATLAB and Excel was found to be the same, which is 897.171.

3.6.5. ACO Method Model Verification

Verification was conducted by comparing the results from running the model in MATLAB with the calculations in Excel. The calculated total distance from both MATLAB and Excel was found to be the same, which is 989.603.

3.7. Comparison between Algorithms

The results obtained from SA, LNS, ALNS and ACO demonstrate clear differences, reinforcing why the suggested layouts are superior. Based on the calculations conducted using three different algorithms, namely SA, LNS, and ALNS, the optimal results obtained were the same. First, SA, LNS, and ALNS provided a more optimal solution, achieving a total material handling distance of 897.171 meters, while ACO resulted in a higher distance of 989.603 meters. This substantial difference in distance, amounting to 92.432 meters, has direct implications for operational efficiency and material handling costs. By minimizing the distance that materials need to be transported, the layout generated by SA helps reduce associated costs, making it a more cost-effective solution.

Additionally, SA reached convergence after 46 iterations, indicating faster stability in finding the optimal layout. On the other hand, while ACO converged earlier at 11 iterations, it yielded a less optimal solution. This disparity can be attributed to the different approaches of these algorithms: SA, LNS, and ALNS effectively balances exploration (searching new areas of the solution space) and exploitation (refining the best-known solutions), leading to more efficient optimization. On the other hand, larger total distance obtained from the ACO algorithm may be due to the lack of balancing between exploration and exploitation capabilities in that algorithm, as ACO is designed with a tendency to prioritize exploration. ACO's tendency to prioritize exploration over exploitation may result in slower convergence to optimal solutions, particularly in constrained environments like the single-row layout problem. ACO's performance may also be highly sensitive to parameters like pheromone evaporation and deposition rates. Therefore, ACO often results in suboptimal solutions, particularly in constrained environments like the single-row layout problem.

The suggested layouts—7-5-6-4-3-2-1 and 1-2-3-4-6-5-7—produced by SA, LNS, and ALNS consistently yielded lower material handling distances than ACO's layout (5-7-6-4-3-2-1). This clear advantage highlights the superiority of the SA, LNS, and ALNS generated layouts, as they provide more efficient material handling paths, reduce operational costs, and improve overall productivity. The combination of lower distance, faster convergence, and a balanced optimization approach makes the layouts suggested by SA superior to those derived from ACO.

Thus, if we only consider the minimum total distance as the objective function, all three algorithms (SA, LNS, and ALNS) are suitable for solving similar problems. However, if we also consider computational time, the SA algorithm is the best option. Based on the average computational time calculated from 10 replications, the SA algorithm shows the fastest average computational time compared to the other algorithms, which is approximately 0.025 seconds.

In addition to the significant cost savings achieved through the reduction in material handling distance, the improvements resulting from the SA, LNS, and ALNS generated layouts can greatly enhance overall efficiency in terms of productivity and workflow. By minimizing the distance materials need to travel across the production floor, the optimized layouts reduce the time required for material movement between workstations. This not only speeds up production cycles but also allows workers to focus more on core tasks rather than handling materials, leading to improved operator performance.

The reduction in material handling also leads to smoother workflows, as there is less disruption caused by frequent or inefficient material transport. With shorter and more direct material paths, production processes can be better synchronized, leading to fewer delays and a more seamless progression of tasks through the different stages of manufacturing. This streamlined flow reduces bottlenecks, which can improve throughput and allow for the handling of more production orders in the same amount of time.

Furthermore, a more efficient layout can improve worker safety and reduce fatigue by limiting the amount of unnecessary movement around the shop floor. By placing machines and workstations in closer proximity, with better alignment to the production process, operators are less likely to encounter obstacles or experience delays, which enhances their focus and contributes to a safer, more productive environment. Therefore, beyond the evident cost reductions, these improvements can lead to increased productivity, optimized workflow efficiency, and a more productive and safer working environment.

4. Conclusion

Based on the search results for optimal production layout using metaheuristic methods with SA, LNS, and ALSN algorithms. A total layout distance of 897,171 meters is obtained using SA and with the optimal machine sequence of 7-5-6-4-3-2-1 and 1-2-3-4-6-5-7. However, when using the ACO algorithm, the total layout distance obtained is 989.603 with a layout of 5-7-6-4-3-2-1. Therefore, the related company needs to modify its facility layout with the facility sequence of 7-5-6-4-3-2-1 or 1-2-3-4-6-5-7 to minimize the total distance of material handling, which is directly related to the material handling cost for the company. The results indicated that the metaheuristics could solve the facility layout problem in sheet metal working industry in efficient manner. In the current study, the primary focus of optimization was on minimizing total material handling distance. While this is a crucial factor in improving facility efficiency, real-world manufacturing environments often require the consideration of additional metrics that can further enhance the effectiveness of the layout.

Therefore, future studies may consider the inclusion of throughput, energy consumption, and flexibility to handle product changes in the layout development. Future works could also explore combining metaheuristic methods to leverage the strengths of multiple approaches and improve performance. In addition, investigating advanced parameter tuning methods, such as adaptive or self-tuning techniques, could also improve the efficiency and effectiveness of the metaheuristics, addressing its current limitations in performance. At last, to broaden the applicability of the findings, future research could test these algorithms in different production settings, such as assembly lines or process industries, with varied constraints like space limitations, safety requirements, or energy consumption. Future research could also explore the applicability of the proposed methods in more complex layout designs such as multi-row or U-shaped layouts, and in various scenarios with larger problem sets.

Author Contribution: All authors contributed equally to the main contributor to this paper. All authors read and approved the final paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

Agista, A. B., Natuna, A. P., Wangsa, H. B., Fernanda, J., Akmal, N. N., & Rifai, A. P. (2021). Perancangan Tata Letak Fasilitas UKM Kerajinan Kayu Dengan Metode Simulated Annealing. *Journal of Industrial and Manufacture Engineering*, *5*(2), 137-147, https://doi.org/10.31289/jime.v5i2.5673.

Alberto, G. C., & Geoff, H. W. (1998). Facility layout overview: towards competitive advantage. *Facilities*, *16*(7-8), 198-203, https://doi.org/10.1108/02632779810221262.

Amaral, A. R. (2006). On the exact solution of a facility layout problem. *European Journal of operational research*, *173*(2), 508-518, https://doi.org/10.1016/j.ejor.2004.12.021.

Besbes, M., Zolghadri, M., & Affonso, R. C. (2021). A method to solve 2D Facility Layout Problem with equipment inputs/outputs constraints using meta-heuristics algorithms. *Procedia CIRP*, *104*, 1698-1703, https://doi.org/10.1016/j.procir.2021.11.286.

Dehghan-Sanej, K., Eghbali-Zarch, M., Tavakkoli-Moghaddam, R., Sajadi, S. M., & Sadjadi, S. J. (2021). Solving a new robust reverse job shop scheduling problem by meta-heuristic algorithms. *Engineering Applications of Artificial Intelligence*, *101*(February), 104207, https://doi.org/10.1016/j.engappai.2021.104207.

Gandhi, H. K., & Widyawati, W. (2019). Penyelesaian Assignment Problem Dengan Algoritma Metaheuristik Ant Colony Optimization (Aco). *Journal Industrial Servicess*, *5*(1), 16–22, https://doi.org/10.36055/jiss.v5i1.6494.

Heragu, S. S., & Kakuturi, S. R. (1997). Grouping and placement of machine cells. *IIE transactions*, *29*(7), 561-571, https://doi.org/10.1080/07408179708966366.

Isnaini, W., Rifai, A. P., Nurmasari, N. M. E., Masruroh, N. A., Dharma, I. G. B., & Andriani, V. E. (2024). Sequential use of blocplan, solver, and particle swarm optimization (PSO) to optimize the double row facility layout. *International Journal of Production Management and Engineering*, *12*(2), 117-124, https://doi.org/10.4995/ijpme.2024.20061.

Kusumaningsih, D. A., Azim, A. F., Albab, D. A. I. U., Hans, F. R., Korin, F., Pohan, R. N. A. M., Ananta, V. S., & Rifai, A. P. (2022). Simulated Annealing untuk Perancangan Tata Letak Industri Furniture dengan Model Single dan Double Row Layout. *Jurnal Media Teknik Dan Sistem Industri*, *6*(1), 60, https://doi.org/10.35194/jmtsi.v6i1.1773.

Liu, S., Zhang, Z., Guan, C., Liu, J., & Dewil, R. (2021). Mathematical formulation and a new metaheuristic for the constrained double-floor corridor allocation problem. *Journal of Manufacturing Systems*, *61*, 155-170, https://doi.org/10.1016/j.jmsy.2021.08.013.

Maghfiroh, M. F. N., Redi, A. A. N. P., Ong, J., & Fikri, M. R. (2023). Cuckoo search algorithm for construction site layout planning. *IAES International Journal of Artificial Intelligence, 12(2),* 851-860, https://doi.org/10.11591/ijai.v12.i2.pp851-860.

Maier, K., & Taferner, V. (2023). Solving the constrained single-row facility layout problem with integer linear programming. *International Journal of Production Research*, *61*(6), 1882-1897, https://doi.org/10.1080/00207543.2022.2051090.

Nugracia, R., & Muslim Lhaksmana, K. (2020). Implementasi Algoritma Simulated Annealing Terhadap Rute Perjalanan Pada Sistem Rekomendasi Objek Wisata. *Q*, *7*(1), 2444, https://openlibrarypublications.telkomuniversity.ac.id/index.php/engineering/issue/view/126.

Pisinger, D., & Ropke, S. (2019). Large neighborhood search. *International Series in Operations Research and Management Science*, *272*, 99–127, https://doi.org/10.1007/978-3-319-91086-4_4.

Retnowati, D., & Fudhla, A. F. (2013). Re-Layout Fasilitas Produksi Industri Sheet Metal Working Berbasis Job Shop Dengan Pendekatan Linear Mix Integer Programming. *Seminar Nasional Manajemen Teknologi XVII*, Surabaya, Indonesia.

Rifai, A. P., Alfithra, A. R., Al'adiat, C. F., & Sari, W. P. (2023). Facility Layout Planning for Pyrolyzer Production Using Automated Layouts Design Program (ALDEP) Method. *OPSI*, *16*(1), 165-173, https://doi.org/10.31315/opsi.v16i1.8745.

Rifai, A. P., Kusumaningsih, D. A., Syahrizad, A., Adriani, A., Hediandra, F. B., Ramadhana, I., Sunarso, R. V. P., & Abdilah, S. H. (2023). Perancangan Tata Letak Fasilitas Industri Bakery dengan Pendekatan Model Single Row dan Double Row Layout. *Jurnal Penelitian dan Aplikasi Sistem & Teknik Industri (PASTI)*, *17*(1), 1-12, https://doi.org/10.22441/pasti.2023.v17i1.001.

Rifai, A. P., Mara, S. T. W., Ridho, H., & Norcahyo, R. (2022). The double row layout problem with safety consideration: a two-stage variable neighborhood search approach. *Journal of Industrial and Production Engineering*, *39*(3), 181-195, https://doi.org/10.22441/pasti.2023.v17i1.001.

Rifai, A. P., Mara, S. T. W., Kusumastuti, P. A., & Wiraningrum, R. G. (2020). A Genetic Algorithm for the Double Row Layout Problem. *Jurnal Teknik Industri*, *22*(2), 85-92, https://doi.org/10.9744/jti.22.2.85-92.

Tompkins, J. A., & Reed Jr, R. U. D. D. E. L. L. (1976). An applied model for the facilities design problem. *The International Journal Of Production Research*, *14*(5), 583-595, https://doi.org/10.1080/00207547608956377.

Tongur, V., Hacibeyoglu, M., & Ulker, E. (2020). Solving a big-scaled hospital facility layout problem with meta-heuristics algorithms. *Engineering Science and Technology, an International Journal*, *23*(4), 951-959, https://doi.org/10.1016/j.jestch.2019.10.006.

Udjulawa, D., & Oktarina, S. (2022). Penerapan Algoritma Ant Colony Optimization Untuk Pencarian Rute Terpendek Lokasi Wisata. *Klik - Jurnal Ilmu Komputer*, *3*(1), 26–33, https://doi.org/10.56869/klik.v3i1.326.