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

Jurnal Optimasi Sistem Industri

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

Research Article

A Multi-Objective Double Row Layout Problem Considering Safety Distances and Geometrical Constraints

Achmad Pratama Rifai [1,*], Setyo Tri Windras Mara [2], Rachmadi Norcahyo [1], Andri Nasution [1,3]

[1] Department of Mechanical and Industrial Engineering, Universitas Gadjah Mada, Special Region of Yogyakarta, Indonesia
[2] School of Systems and Computing, University of New South Wales, Canberra, Australia
[3] Department of Industrial Engineering, Universitas Sumatera Utara, Medan, Indonesia

*Corresponding Author: achmad.p.rifai@ugm.ac.id

DOI: 10.25077/josi.v24.n2.p174-197.2025 **Submitted:** August 7, 2025 **Accepted:** November 28, 2025 **Published:** December 30, 2025

ABSTRACT

This study addresses an enhanced version of the Double Row Layout Problem (DRLP) by incorporating two critical constraints: minimum safety distances between machines and geometric limitations on row lengths. A bi-objective mixed-integer non-linear programming (MINLP) model is formulated to simultaneously minimize material handling costs and penalties for violating safety distance requirements. To solve the problem efficiently, a novel metaheuristic called Improved Multi-Objective Variable Neighborhood Search (IMOVNS) is proposed. IMOVNS extends the standard MOVNS by integrating an adaptive archive update strategy and a probabilistic acceptance mechanism inspired by AMOSA, thereby improving both convergence and diversity in Pareto front generation. This study contributes to the layout optimization literature by proposing a tailored MOVNS variant explicitly designed for safety-aware and geometry-constrained DRLP, a challenging problem variant that has received limited attention in prior research. Extensive experiments on 27 DRLP instances show that IMOVNS demonstrates strong performance, significantly outperforming NSGA-II and showing competitive or superior results compared to AMOSA and MOVNS in terms of convergence and solution diversity. Statistical tests further confirm the significant superiority of IMOVNS, particularly over NSGA-II. Additionally, a key managerial insight reveals that layouts with unbalanced row lengths favour safety compliance, while balanced layouts minimize material handling costs. The Pareto-optimal solutions generated by IMOVNS enable decision-makers to select layout configurations that align with specific operational priorities. These findings highlight the practical relevance and robustness of IMOVNS in solving real-world multi-objective facility layout problems under complex spatial and safety constraints.

Keywords: double row layout problem, multi-objective optimization, safety distance, geometric constraint.

INTRODUCTION

In the context of Industry 4.0 and lean manufacturing, arranging production equipment efficiently has become a critical concern for improving system productivity, operational flexibility, and environmental performance. The Facility Layout Problem (FLP) plays a major role in this effort, as well-designed layouts help eliminate unnecessary movements, increase the utilization of resources, and significantly lower material handling expenses costs [1]. According to Mohamadghasemi and Hadi-Vencheh [2], material handling costs can represent as much as 20–50% of total operating costs. Earlier research by Liu et al. [3] have shown that that well-structured layouts are capable of reducing these expenses by approximately 10–20%.

Among the different variations of FLP, the Double Row Layout Problem (DRLP) has received increasing interest due to its closer alignment with real manufacturing practices. The DRLP builds on the traditional Single Row Layout

Problem (SRLP), which focuses on arranging facilities along a single handling aisle [4]. The DRLP, first introduced by Chung and Tanchoco [5], provides a more realistic representation of shop-floor conditions by placing machines on both sides of the aisle. This modification expands the range of feasible solutions and introduces greater complexity to the problem.

This work extends the DRLP by incorporating two important elements commonly found in practice: (1) minimum safety distances between machines and (2) limitations arising from unequal row lengths. In many industrial environments, machines cannot be placed too closely together because of safety risks, such as fire hazards, noise, vibrations, or emissions [6]. However, these safety considerations are not included in the original DRLP formulation. Moreover, floor space constraints frequently lead to layouts where each row has a different available length [7]; [8].

To address this enriched version of the problem, a multi-objective optimization model is developed with two competing objectives: minimizing total material handling cost and minimizing penalties related to safety violations. To efficiently solve this model, we introduce an Improved Multi-Objective Variable Neighborhood Search (IMOVNS), which builds upon the existing MOVNS framework [9]; [10], by integrating a solution archive and a Metropolis-style acceptance rule [11].

The Double Row Layout Problem (DRLP) was first proposed by Chung and Tanchoco [5], who formulated the problem as a mixed-integer linear program (MILP). who modeled the arrangement of machines along two opposing rows using a mixed-integer linear programming (MILP) framework. They compared five heuristic construction methods—MinLCF (minimum location cost first), MinFF (minimum flow first), MaxFF (maximum flow first), MinWF (minimum width first), and MaxWF (maximum width first)—to generate promising initial layouts. Due to the combinatorial complexity involved, even instances with only ten machines required several hours to solve optimally.

Subsequent studies aimed to strengthen and generalize the initial DRLP formulation. Zhang and Murray [12] modified the original model by incorporating explicit clearance requirements to reflect varying safety distances between workstations. Murray et al. [13], later extended this work by developing a mixed-integer linear DRLP (MILD) model that accommodates asymmetric material flows. Their approach combined eight constructive heuristics with a local search strategy, resulting in noticeable performance improvements in solution quality. More recently, Amaral [14] introduced a more compact MILP model with a reduced number of binary and continuous decision variables, making the formulation easier to interpret and solve. Secchin and Amaral [14] further strengthened this model by adding new constraints related to machine placement, improving the tightness of the formulation.

Research on facility layout problems (FLPs) has evolved considerably over time, with a strong emphasis on reducing material handling costs while arranging machines and departments efficiently [15]; [16]. Early FLP models typically assumed simple geometric forms and omitted spatial or safety-related limitations. As shop-floor space has become increasingly restricted, new extensions emerged, including the DRLP introduced by Chung and Tanchoco [5] which reflects a more realistic need to place machines in two parallel rows under limited floor capacity.

The initial DRLP formulation by Chung and Tanchoco [5] focused solely on minimizing material handling costs through mixed-integer programming. Amaral [14] later refined this formulation, simplifying the objective and improving computational tractability. However, this remained a single-objective perspective. Chae and Regan [17] expanded the scope by incorporating geometric layout constraints and heuristic rules to generate layouts that are more feasible in practice.

More recent developments have progressed toward multi-objective DRLP and related FLPs, explicitly incorporating competing considerations such as spatial utilization and energy efficiency [18]; [19]; [20]. Various review articles

 DOI: 10.25077/josi.v24.n2.p174-197.2025

also document the shift toward multi-criteria models [21]. Nevertheless, only a small number of DRLP studies explicitly consider minimum safety distances or unbalanced row dimensions [22], leaving a methodological need for approaches that manage safety, geometry, and operational efficiency simultaneously.

Owing to the high computational difficulty of DRLP, many researchers have explored heuristic or metaheuristic approaches. Rifai et al. [22] proposed an Improved Variable Neighborhood Search (IVNS) tailored to DRLP with safety distance requirements, ensuring minimum spacing between machines to avoid hazardous proximity. Isnaini et al. [23] employed a Particle Swarm Optimization (PSO) approach to address the DRLP in the context of train manufacturing. Their method used BLOCPLAN as a construction heuristic to generate starting layouts, substantially enhancing the performance of PSO when compared with random initialization.

A number of recent studies have adopted multi-objective formulations to better capture trade-offs embedded in DRLP. Zuo et al. [24] examined the asymmetric DRLP through a Multi-Objective Tabu Search method that simultaneously considered transportation costs and layout area as conflicting criteria. Their framework relied on linear programming to determine non-dominated continuous layouts and applied Tabu Search for further refinement. It achieved faster computation and competitive layout quality when compared with exact methods and NSGA-II. Similarly, Wang et al. [25] addressed a multi-objective DRLP with a secondary objective to minimize layout area under conditions where aisle width is non-zero. Their model was solved through a hybrid approach combining mathematical programming and Improved Simulated Annealing (ISA).

Algorithmic research on DRLP has steadily moved from exact optimization toward metaheuristic techniques, including genetic algorithms, simulated annealing, and variable neighborhood search, with the aim of improving scalability and solution quality for increasingly large problem instances. However, most existing metaheuristic methods are still limited in their ability to simultaneously handle safety constraints or geometric asymmetry. In addition, adaptive, archive-based multi-objective schemes, such as AMOSA or MOVNS, remain largely unexplored in DRLP applications. Addressing this gap is a key motivation of the present study.

In conventional FLPs, material handling cost is often the primary objective and is modelled as a function of flow volumes, movement costs, and rectilinear distances between departments. Aiello et al. [6] investigated unequal-area FLPs by considering four objectives: aspect ratio, material handling cost, closeness ratings, and interdepartmental distances, using NSGA-II for multi-objective optimization. The ELECTRE decision framework was then used to rank solutions based on managerial preferences. Jolai et al. [26] introduced a Multi-Objective PSO (MOPSO) model to jointly minimize rearrangement and handling costs while maximizing adjacency and proximity relations. Their findings demonstrated the suitability of MOPSO for layouts involving departments of unequal sizes. Ingole and Singh [7] applied the Firefly Algorithm (FA) to fixed-shape FLPs to reduce transportation and handling costs. The FA delivered better solutions than hybrid GA, standard GA, and Tabu Search while requiring less computation time.

Further improvements to metaheuristics have emerged, including Discrete PSO[27], which maps binary decisions to layout configurations suitable for dynamic FLP environments. Derakhshan Asl et al. [28] implemented an enhanced Covariance Matrix Adaptation Evolution Strategy (CMA-ES) for FLPs with unequal department areas, internal walls, and geometric constraints, outperforming both enhanced PSO and GA. RazaviAlavi and AbouRizk [29] used Genetic Algorithms for tunnelling site layouts, incorporating departmental sizes and interdepartmental distances to guarantee physical feasibility under space limitations. Although these studies successfully consider several practical elements, no existing work has addressed DRLP under conditions where row lengths are unequal and safety separations are required. In actual industrial environments, the combined influence of safety requirements and limited space leads to complex trade-offs that existing DRLP frameworks are not designed to manage.

To fill this methodological gap, this study proposes an Improved Multi-Objective Variable Neighborhood Search (IMOVNS), extending the classical Variable Neighborhood Search (VNS) introduced by Mladenović and Hansen

[30]. The VNS concept has been widely applied to combinatorial problems such as the Single Row Layout Problem [1], multi-level warehouse layouts [31], and multi-objective FLPs with unequal areas [32].

The contributions of this study are threefold. First, we introduce geometric constraints that reflect real shop-floor conditions in which machine locations are affected by the irregular availability of space due to adjacent facilities such as AGV docks, storage zones, or offices. These spatial limitations naturally result in unequal row lengths, which is an aspect rarely considered in DRLP literature. Second, this research examines how safety distance constraints interact with spatial asymmetry and how these interactions influence layout feasibility and performance. This integrated treatment reveals the interplay between material handling efficiency, geometric feasibility, and safety compliance. Third, we develop a multi-objective metaheuristic tailored to industrial-scale DRLP instances. Although metaheuristics have been applied to FLPs more broadly, their use in DRLP settings remains limited. The proposed approach contributes to expanding their applicability for complex layout planning in double-row configurations.

METHODS

Model Formulation

This section describes the mathematical model developed for the Double Row Layout Problem (DRLP), incorporating both safety distance and geometric constraints. It is followed by an overview of the optimization strategy employed in the study. The DRLP seeks to arrange machines or departments across two parallel rows in a way that minimizes overall material handling cost while ensuring that minimum safety clearances between machines are respected. Each machine has a specified width, and the total allowable length of each row is limited by available floor space. The aim is to produce feasible layouts that satisfy spatial restrictions and safety requirements, while also reducing the selected objective costs. Figure 1 illustrates the general configuration of a double row layout in which machines are placed along two parallel rows with flows occurring between them. This layout serves as the basis for formulating the optimization model.

In the context of this study, we consider a specific variant of the DRLP, which incorporates geometry constraints and safety zones that affect layout flexibility. The problem considered comprises of the arrangement of a set of m machines in a corridor with a row on both sides. We consider the situation where each pair of machines i and j is a subject of a recommended safe distance ϕ_{ij} and a penalty may incur when this recommendation is violated. Each row k also has a hard constraint of length which denoted by V_k. This results in a possibility of having two rows with unequal length, which represent the potential condition in a real-life situation.

Figure 1 shows an instance of the Double Row Layout Problem (DRLP) under geometric constraints. In this case, the positions of seven machines must be determined across the two rows. The available placement area for each machine is indicated by the dotted line. It is worth noting that the upper and lower rows differ in length due to the presence of facilities A and B, which serve non-machining purposes such as material handling, parking, or engineering office functions.

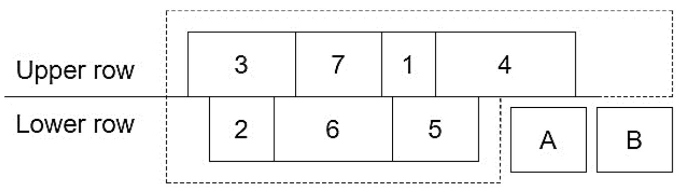


Figure 1. Illustration of DRLP with geometry constraint

 DOI: 10.25077/josi.v24.n2.p174-197.2025

Table 1. Notations, Sets, Parameters, and Variables

Notations	
i, j	Index of machine
k	Index of row
m	Number of machines
Sets	
I	Set of machines, $I = \{1, \ldots, m\}$
K	Set of rows, $K = \{1,2\} = \{\text{Upper row}, \text{Lower row}\}$
Parameters	
l_i	Length of machine i
L	Sum of the length of all machines, $L = \sum_{i \in N} l_i$
f_{ij}	Material flow between a pair of machines i and j
ϕ_{ij}	Recommended safe distance between a pair of machines i and j
V_k	Length of row k
M	A large number
Variables	
d_{ij}	Distance between a pair of machines i and j
d_{ij}^{+}	Dummy variable to capture the distance between machines i and j when machine i is on the left side of machine j
d_{ij}^{-}	Dummy variable to capture the distance between machines i and j when machine i is on the right side of machine j

We develop a Mixed-Integer Non-Linear Programming (MINLP) model based on the original DRLP formulation by Chung and Tanchoco [5], as the more recent formulations by Amaral [14] and Chae and Regan [17] do not explicitly specify the row assignment of machines. The objectives of the proposed model are to simultaneously:

(i) minimize the total material handling cost, and

(ii) minimize the penalty due to safety distance violations.

The first objective is computed as the product of material flow and the Euclidean distance between each machine pair. The second objective is derived from the magnitude of the safety distance violations between machines. To formulate the model, we first define the necessary notations, sets, parameters, and decision variables, as listed in Table 1, including the following:

x_{ik} = The abscissa location of machine i in row k, the value is 0 if the given machine is not located at row k

y_{ik} = Binary variable to indicate the row location of a machine. It takes the value of 1 if machine i is placed at row k, and 0 otherwise.

z_{ijk} = Binary variable to indicate the abscissa location of a machine in regard to another machine in the same row. It takes the value of 1 if machine j is positioned on the right side of machine i in row k, and 0 otherwise.

Then, the following MINLP formulation is proposed as follow.

$$\min f_1 = \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} f_{ij} d_{ij} \quad (1)$$

$$\min f_2 = \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} \max(0, \phi_{ij} - d_{ij}) \quad (2)$$

subject to:

$$x_{ik} \leq M y_{ik} \qquad \forall\, i \in I, k \in K \qquad (3)$$

$$\sum_{k=1}^{K} y_{ik} = 1 \qquad \forall\, i \in I \qquad (4)$$

$$\frac{l_i y_{ik} + l_j y_{jk}}{2} \leq x_{ik} - x_{jk} + M z_{ijk} \qquad \forall\, i \in I, j \in I, i < j, k \in K \qquad (5)$$

$$\frac{l_i y_{ik} + l_j y_{jk}}{2} \leq -x_{ik} + x_{jk} + M(1 - z_{ijk}) \qquad \forall\, i \in I, j \in I, i < j, k \in K \qquad (6)$$

$$\sum_{k=1}^{K} x_{ik} - \sum_{k=1}^{K} x_{jk} + d_{ij}^{+} - d_{ij}^{-} = 0 \qquad \forall\, i \in I, j \in I, i < j \qquad (7)$$

$$d_{ij} = \max\,(d_{ij}^{+}, d_{ij}^{-}) \qquad \forall\, i \in I, j \in I, i < j \qquad (8)$$

$$x_{ik} + \frac{l_i}{2} \leq V_k \qquad \forall\, i \in I, k \in K \qquad (9)$$

$$d_{ij}^{+}, d_{ij}^{-} \geq 0 \qquad \forall\, i \in I, j \in I, i < j \qquad (10)$$

$$x_{ik} \geq 0 \qquad \forall\, i \in I, k \in K \qquad (11)$$

$$y_{ik} \in \{0,1\} \qquad \forall\, i \in I, k \in K \qquad (12)$$

$$z_{ijk} \in \{0,1\} \qquad \forall\, i \in I, j \in I, i < j, k \in K \qquad (13)$$

The proposed MINLP involves two objective functions which subjected to a set of constraints. The objective function (1) is set to minimize the total material handling cost within the corridor. The objective function (2) aims to minimize the penalty for safety factor which calculated when the distance between a pair of machines i and j (d_{ij}) violates the recommended safety distance ϕ_{ij}. Equation (3) ensures that the abscissa variable x_{ik} is only active when the machine i is located on the corresponding row k. Equation (4) guarantees that a given machine i is placed only in a single row. Equations (5)-(6) prevent the overlap between a pair of machines when they are in a same row k. Equations (7)-(8) calculates the distance between machines i and j. Equation (9) is the geometric constraint which avoids the violation of space limitation in each row k. Lastly, Equations (10)-(11) are the integrality constraints while Equations (12)-(13) limit the value of binary decision variables.

The Proposed Metaheuristics

This section describes the proposed method for obtaining the non-dominated solutions for the multi objective DRLP in detail.

IMOVNS Algorithm

The Variable Neighborhood Search (VNS), first introduced by Mladenović and Hansen [30], is constructed based on a systematic change among predefined neighborhood structures that guide the search toward a local optimum. It also includes a perturbation mechanism commonly referred to as shaking to enable the algorithm to escape from local valleys in the solution landscape [33]. The core concept of this metaheuristic lies in two key observations:

(i) a global minimum can be viewed as a local minimum with respect to all possible neighborhood structures, and
(ii) local minima from various neighborhood structures in many problems tend to be located in close proximity to one another [34].

Arroyo et al. [9] extended the original VNS for multi-objective optimization by proposing the Multi-Objective Variable Neighborhood Search (MOVNS). Building upon this, the present study enhances MOVNS by incorporating a solution acceptance criterion and archive update mechanism adapted from the Archived Multi-Objective Simulated Annealing (AMOSA) approach [35].

 DOI: 10.25077/josi.v24.n2.p174-197.2025

Input: an instance (n, l, d, ϕ, c), VNS parameters
$(\Gamma_1, \Gamma_0, \alpha, K, Q_1, Q_0)$
initialize neighborhood list $S = \{1, \dots, |S|\}$
Initialize feasible solutions x
calculate each fitness value $f_o(x),\ o \in O$
insert x to ND
while $\Gamma_c \geq \Gamma_1$ **do**
  $x' \leftarrow \text{shaking}(x, s, S, Q_1, Q_0)$
  $x'' \leftarrow \text{Randomized VND}(x, f(x), x', S, n, l, d, \phi, c)$
  If acceptance criteria are fulfilled, **then**
    $x \leftarrow x''$
  end if
  update ND
end while
Output: $ND, f_o(s^{ND})\ \ \forall\, o \in O$

Figure 2. Pseudocode of the IMOVNS Algorithm

An extensive set of tailored heuristics is proposed to modify solutions, with specific operators developed to promote both intensification and diversification of the search. This design aims to effectively balance the trade-off between exploration and exploitation within the solution space. The pseudocode of the proposed Improved Multi-Objective Variable Neighborhood Search (IMOVNS) for solving the DRLP is summarized in Figure 2.

Let us consider a combinatorial optimization problem where the objective is to find a solution point x within the solution space X, such that the value of the objective function $f(x)$ is minimized. The algorithm begins by initializing a solution x that is generated stochastically. Subsequently, the fitness of this solution is evaluated, where O denotes the set of objective functions, and the solution $x \in X$ is stored in the non-dominated solution archive ND. Constraint-handling is applied during the evaluation stage.

A neighborhood list $S = \{1, \dots, |S|\}$ is then initialized, consisting of sss neighborhood structures that are employed in the proposed framework. In each iteration, three distinct steps are executed. First, a shaking procedure is applied to perturb the current solution x using a move selected from the neighborhood N_n. The resulting perturbed solution x' becomes the starting point for the second step: a local search heuristic, in which multiple neighborhoods are explored to generate a new solution x''. If the newly generated solution x'' is non-dominated, it is adopted as the current solution and added to the non-dominated solution archive ND. Otherwise, an acceptance mechanism is invoked to decide whether to retain the current solution from among the new solution x″, the previous solution x, or a randomly selected solution from the archive ND. Following this, the archive ND is updated by removing any solutions that become dominated due to the inclusion of the new non-dominated solution. This iterative process continues The iteration continues while the current temperature Γ_c remains greater than the final temperature Γ_1.

Solution Representation

The multi-objective Double Row Layout Problem (DRLP) focuses on assigning a set of machines into two rows under geometric constraints, such that both the material handling cost and the safety penalty are minimized. In this study, a two-row representation scheme is adopted, as illustrated in Figure 3. The first row denotes the row assignment of each machine, while the second row indicates the machine index within its allocated row.

Machine i	1	2	3	4	5	6	7
Row	1	2	1	1	2	2	1
Machine Index	3	1	1	4	3	2	2

Figure 3. Solution representation for a feasible double-row layout with n=7 machines

This type of representation enables the solution perturbation process to operate in two modes:

(i) modifying the machine sequence across both rows to facilitate broader exploration, and

(ii) rearranging machines within a specific row to support local refinement and exploitation.

As shown in Figure 3, this representation corresponds to a feasible layout configuration with n=7, derived from the example illustrated in Figure 1. Let the set of machines be denoted by $N = \{1,2,3,\ldots,n\}$, where a certain amount of product flow is required between machines.

The **first row** in the representation assigns each machine to one of the two layout rows: $r_1 = \{i_1, i_3, i_4, i_7\}, r_2 = \{i_2, i_5, i_6\}$. The **second row** defines the position sequence of machines within their respective rows. For instance, the machine sequence for row r_1 from the starting point *e* is: $i_3 \rightarrow i_7 \rightarrow i_1 \rightarrow i_4$. Importantly, the sequence numbers for machines within the same row are always unique to ensure that no machine placement is duplicated, thereby prohibiting redundancy. In the actual computational implementation, the solution representation used by the algorithm consists solely of the row allocation vector *r* and the machine index vector (i.e., the positional sequence), which are visually highlighted in light grey. The machine codes *i* are omitted during computation, as their identities are fixed and do not affect the layout decision process.

The corresponding x-coordinate is computed deterministically: the first machine in each row is placed with its left edge at $x = 0$, and the subsequent machine i is positioned at $x_i = \sum_{j=1}^{i-1}(w_j + s_j)$, where w_j denotes the width of machine *j* and s_j is the safety distance between adjacent machines. For example, if three machines with widths of 3, 4, and 2 units are arranged in sequence with 1-unit safety gaps, their left-edge coordinates are $x = 0$, $x = 4$, and $x = 9$, respectively.

Shaking Procedure

In order to explore the numerous possibilities of row allocations and machine order in each row, the proposed algorithm is equipped with four specially-tailored neighborhood structures for shaking. These are: (i) swap inter-row, (ii) insertion inter-row, (iii) reverse inter-row, and (iv) row change moves. The procedure and illustration of the moves are visually presented in Figure 4 in the respective order.

The first and second moves can be considered as a move with low magnitude. Let M_{ilk} represents the machine i at sequence l of row k. Two machines M_{ilk} and M_{jmp} where $k, p \in R$ and $k \neq p$ are selected and the location of these machines (nodes l and m, respectively) are swapped. Similarly, in the insertion inter-row move, machines M_{ilk} and M_{jmp} are randomly selected where $k \neq p$ and $m \neq 0$. Afterwards, the machine M_{ilk} is placed just before the machine M_{jmp} so that $l = m, m' = m + 1$, and $k = p$.

The third neighborhood can be seen as a move with a larger magnitude than the previous two. Thus, these moves are logically presented in the higher orders than the family of swap and insertion moves. The reverse inter-row move selects M_{ilk} and M_{jmp}, then all machines that fall in between M_{ilk} and M_{jmp} are represented in a reversed order. Note that when $i - j = 1$ holds true, the reverse inter-row move is a special case of the swap move. The row change

 DOI: 10.25077/josi.v24.n2.p174-197.2025

Machine i	1	2	3	4	5	6	7
Row	1	2	1	1	2	2	1
Machine Index	3	1	1	4	3	2	2

1	2	3	4	5	6	7
1	1	1	2	2	2	1
3	4	1	1	3	2	2

a) Swap inter-row

Machine i	1	2	3	4	5	6	7
Row	1	2	1	1	2	2	1
Machine Index	3	1	1	4	3	2	2

1	2	3	4	5	6	7
2	2	1	1	2	2	1
2	1	1	4	4	3	2

b) Insertion inter-row

Machine i	1	2	3	4	5	6	7
Row	1	2	1	1	2	2	1
Machine Index	3	1	1	4	3	2	2

1	2	3	4	5	6	7
1	2	2	2	1	1	1
3	1	2	3	4	1	2

c) Reverse inter-row

Machine i	1	2	3	4	5	6	7
Row	1	2	1	1	2	2	1
Machine Index	3	1	1	4	3	2	2

1	2	3	4	5	6	7
2	2	2	1	2	1	1
3	1	2	3	4	1	2

d) Row-change

Figure 4. Solution representation

move follows different procedures as it simply changes the row of the selected machines. The number of selected machines is determined by the linear alteration schedule [36] described as follow.

$$\Upsilon_c = Q_c\, N \qquad \forall \;\; c \in \{1, 2, \ldots, \mathrm{T}\} \tag{1}$$

$$Q_c = Q_1 - (Q_1 - Q_0)(I_c / I_T) \qquad \forall \;\; c \in \{1, 2, \ldots, \mathrm{T}\} \tag{2}$$

where Υ_c is the number of destroyed elements in the current iteration I_c, and N is the number of jobs (length of the solution). Its value is determined according to destroy ratio Q_c, which subsequently is influenced by the total number of iterations I_T, the pre-determined initial destroy ratio Q_1, and the final destroy ratio Q_0. The destruction ratio follows a linear alteration schedule, chosen for its ability to ensure a controlled and progressive reduction in solution perturbation intensity. This design allows for greater diversification in early stages and gradual intensification as the temperature decreases, leading to more stable convergence behaviour compared to exponential or adaptive schedules.

The four neighborhood moves—swap, insertion, reverse, and row-change—were selected for their complementary search behaviors and successful application in prior facility layout and sequencing studies [22]. Swap and insertion operations promote intensification by exploring local variations of the current layout with low modification magnitude, while reverse and row-change moves introduce exploration by performing larger structural adjustments across or within rows. This combination ensures both local refinement and global diversification of solutions, preventing premature convergence and improving the robustness of the search process.

These four neighborhood moves are executed in the previously mentioned order. The moves for this shaking procedure are aimed for exploration since both of row assignment and the machine index can be modified. Hence,

Input: x, s, S
If $s = 1$, **then:**
    $x' \leftarrow$ swap intra-row (x)
Elseif $s = 2$, **then:**
    $x' \leftarrow$ insertion inter-row (x)
Elseif $s = 3$, **then:**
    $x' \leftarrow$ reverse inter-row (x)
Elseif $s = 4$, **then:**
    $x' \leftarrow$ Row change (x)
Output: x'

Figure 5. Pseudocode of Shaking procedure

the machine can be transferred to another row in the solution modification. The pseudocode of the shaking procedure is presented in Figure 5.

Variable Neighborhood Structure

The Variable Neighborhood Descent (VND) is employed as a strategy for changing neighborhood within the local search phase of INVS algorithm. VND is the primitive version of VNS that was presented by Mladenović and Hansen [30] and can be seen as the basic local search structure that can be employed within the VNS framework when multiple neighborhood structures are considered. In this study, the randomized VND adapted from de Freitas and Penna [37] is developed to enhance the exploitation performance of the algorithm. It differs from the classical VND due to the involvement of shuffle procedure in Step 8 of Figure 6. In this step, when the acceptance criterion is met, the neighborhood list *N* will be rearranged. However, unlike the previous method, this study randomized the number of neighborhoods lists *N* at each iteration.

Three neighborhood structures are deployed as the exploitation moves: (i) swap intra-row, (ii) insertion intra-row, and (iii) reverse intra-row moves. These moves act as refining method by altering the machine placement within a row. The moves in the neighborhood of VND are visually presented in Figure 7.

The moves for VND are similar to their counterparts in the shaking process, with a fundamental difference that the moves for VND are executed for the machines in the same row. In the swap intra-row move, two machines M_{ilk} and

1 **Input:** $x, f(x), x', S, n, l, d, \phi, c$
2 $sMax \leftarrow |S|$
3 $s \leftarrow 1$
4 **While** $s \leq sMax$ **do:**
5   Execute the move in neighborhood S_s for altering x' to yield x''
6   **If** $f(x'') < f(x')$ **then:**
7         $x' \leftarrow x''$
8   Shuffle S
9         $s \leftarrow 1$
10   **Else:**
11         $s \leftarrow s + 1$
12 **Output:** $x', f(x')$

Figure 6. Pseudocode of Randomized Variable Neighborhood Descent

 DOI: 10.25077/josi.v24.n2.p174-197.2025

Machine i	1	2	3	4	5	6	7
Row	1	2	1	1	2	2	1
Machine Index	3	1	1	4	3	2	2

1	2	3	4	5	6	7
1	2	1	1	2	2	1
3	1	2	4	3	2	1

a) Swap inter-row

Machine i	1	2	3	4	5	6	7
Row	1	2	1	1	2	2	1
Machine Index	3	1	1	4	3	2	2

1	2	3	4	5	6	7
1	2	1	1	2	2	1
4	1	1	2	3	2	3

b) Insertion inter-row

Machine i	1	2	3	4	5	6	7
Row	1	2	1	1	2	2	1
Machine Index	3	1	1	4	3	2	2

1	2	3	4	5	6	7
1	2	1	1	2	2	1
1	1	3	4	3	2	2

c) Reverse inter-row

Figure 7. Moves in the neighborhood of VND

M_{jmk} in the same row $k \in \{1,2\}$ where $i, j \in n$ are selected, then their position is swapped. In the reverse intra-row move, two machines M_{ilk} and M_{jmk} in the same row are selected, then all machines in the selected row k that fall in between M_{ilk} and M_{jmk} are rearranged in a reversed order.

Solution Acceptance and Termination Criteria

The acceptance criteria and Archive update are developed based on the dominance analysis of AMOSA [35]. Let s be the current solution, x' be the newly generated solution as a product of destroy and repair operators, and A∈ND be a member of the non-dominated set. Then, three cases may arise, as described in the pseudocode of Figure 8. The variable $\Delta_{x,x'}$ is the amount of domination between solution x and x', and calculated as follows.

$$\Delta_{x,x'} = \prod_{o=1, f_o(x) \neq f_o(x')}^{O} |f_o(x) - f_o(x')| / (R1_o - R2_o) \tag{16}$$

$$\Gamma_c = \alpha\Gamma_{c-1} \tag{17}$$

where $R1_o$ and $R2_o$ are the maximum and minimum values of objective o obtained by the members of the non-dominated set ND. The variables W_1, W_2, and W_3represent weighting factors that balance dominance, diversity, and similarity considerations during acceptance. The parameters $\xi_d$and $\xi_I$are tolerance thresholds for dominance and indifference, respectively. The domination measure Δquantifies the relative improvement between the candidate and reference solutions across normalized objectives. A higher Δvalue indicates stronger dominance. The complete description of this mechanism, including probabilistic acceptance rules and detailed parameter interactions, can be found in reference [29].

In some scenarios, the probability of accepting the worse solution x' is influenced by the current temperature Γ_c, in which its value is updated based on Eqeation (17). Hence, this algorithm may accept worse newly generated solution in the earlier iteration to allow further exploration of the search space, and limit accepting worse solution in later iterations to focus on exploitation process of promising neighborhood. The iteration continues until the current temperature Γ_c reaches the final temperature Γ_1.

```
Input: $x', x, ND, \Gamma_c$
check the domination status of $x'$ with respect to $x$ and $ND$;
if $x \succ x' \land A_k \succ x',\ k \geq 0$
  $\bar{\Delta} = \left(\left(\sum_{i=1}^{k} \Delta_{i,x'}\right) + \Delta_{x,x'}\right)/(k+1)$;
  $P = 1/\left(1 + e^{(\bar{\Delta} \times \Gamma_c)}\right)$;
  set $x \leftarrow x'$ with probability $P$; $\xi_{d,I} = W_3$;
elseif $x$ and $x'$ are non-dominating each other
  if $A_k \succ x',\ k \geq 1$;
    $\bar{\Delta} = \left(\sum_{i=1}^{k} \Delta_{i,x'}\right)/k$;
    $P = 1/\left(1 + e^{(\bar{\Delta} \times \Gamma_c)}\right)$;
    set $x \leftarrow x'$ with probability $P$; $\xi_{d,I} = W_3$;
  elseif all $A \in ND$ and $x'$ are non-dominating each other
    set $x \leftarrow x'$;
    insert $x'$ to $ND$; $\xi_{d,I} = W_2$;
  elseif $x' \succ A_k,\ k \geq 1$
    set $x \leftarrow x'$;
    insert $x'$ to $ND$;
    $ND = ND \setminus A_k$; $\xi_{d,I} = W_1$;
  end if
elseif $x' \succ x$
  if $A_k \succ x',\ k \geq 1$
    $\Delta' = min_{i \in k} \Delta_{i,x'}$;
    $P = 1/\left(1 + e^{(-\Delta')}\right)$;
    $\mathrm{r} = \mathrm{U}(0,1)$;
    if $\mathrm{r} < P$
      set $x \leftarrow x'$; $\xi_{d,I} = W_3$;
    else
      set $x \leftarrow A_k$ with $\Delta'$; $\xi_{d,I} = W_3$;
    end if
  elseif $A \in ND$ and $s'$ are non-dominating each other
    set $x \leftarrow x'$;
    insert $x\,'$ to $ND$; $\xi_{d,I} = W_2$;
  elseif $x' \succ A_k,\ k \geq 1$
    set $x \leftarrow x'$;
    insert $x'$ to $ND$;
    $ND = ND \setminus A_k$; $\xi_{d,I} = W_1$;
  end if
end if
Output: $x, ND$
```

Figure 8. Pseudocode for the acceptance criteria and Archive update

RESULTS AND DISCUSSION

This section presents the experimental results obtained from implementing the proposed IMOVNS method on the Double Row Layout Problem (DRLP). The objective of this experiment is to analyse the solution quality generated by IMOVNS in comparison with benchmark algorithms, namely AMOSA, MOVNS, and NSGA-II.

Test Problems

The test problems used in this study are adapted from the DRLP benchmark datasets provided by Simmons [38], Amaral [39], [40], [41], and Secchin & Amaral [42]. These problems are widely used in literature for evaluating DRLP algorithms. The input includes flow data and dimensions of machines, which already account for minimum

 DOI: 10.25077/josi.v24.n2.p174-197.2025

required distances between machines. All problems assume symmetrical flow, with values ranging from 0 to 999, where zero indicates the absence of flow between a pair of machines. The minimum distance requirements between machines are randomly generated from the interval (0, 1) × machine length, where a value of 0 means no safety distance is required. It means that the assigned minimum distance scales proportionally with the physical size of the machines. This approach preserves geometric realism by relating safety spacing to actual machine dimensions while maintaining variability across test instances. Additionally, geometric constraints are introduced by restricting the maximum layout length in each row to simulate practical limitations.

Evaluation metrics

To assess the quality of the generated non-dominated solutions, this study employs five well-established metrics that collectively evaluate convergence, uniformity, cardinality, distribution, and relative performance across multiple algorithms. In problems such as the Double Row Layout Problem (DRLP), where the true Pareto front is either unknown or computationally intractable, these metrics are computed relative to two reference points: the ideal (perfect) point and the nadir point, defined as follows:

$$f_o^b = \min_{x \in ND,\ ND \in ND'} f_o(x) \quad \forall\, o \in O \tag{18}$$

$$f_o^w = \max_{x \in ND,\ ND \in ND'} f_o(x) \quad \forall\, o \in O \tag{19}$$

Here, $f_o^b \in f^b$ denotes the ideal point, representing the best (minimum) value achieved for objective o across all non-dominated sets ND' generated by the compared algorithms. Conversely, $f_o^w \in f^w$ denotes the nadir point, representing the worst (maximum) value for each objective. All objective values are normalized with respect to these two reference points to ensure comparability and robustness in metric computation.

Specifically, $f_o^b \in f^b$ is the ideal value for objective $o \in O$, while $f_o^w \in f^w$ is its corresponding nadir value. The ideal point is a hypothetical best point, defined as the smallest value for each objective among all combined non-dominated solutions ND' produced by the algorithms. Similarly, the nadir point is a hypothetical worst point, defined as the largest value for each objective among all non-dominated solutions ND'. For evaluation metric calculations, objective values are normalized using these two reference points.

The five metrics used in this study are described as follows, based on the formulations provided by Laszczyk and Myszkowski [43] and Li and Yao [44]:

Average Euclidean Distance (ED): This metric evaluates convergence by computing the average Euclidean distance between the obtained non-dominated solutions and the centre of the normalized objective space. A smaller ED indicates closer proximity to the central region, implying better convergence behaviour.

Purity (P): Purity measures the proportion of solutions produced by a specific algorithm that remain non-dominated when compared against the collective solution set of all algorithms. Higher purity values reflect superior relative performance in preserving globally non-dominated solutions.

Minimum Sum of Objective Values (SO): These metric captures convergence by identifying the solution with the lowest total sum across all objectives. It evaluates how close this best-compromise solution lies relative to the ideal point, providing insight into the overall quality of approximation.

Hypervolume (HV): HV quantifies both convergence and diversity by calculating the volume in the objective space that is dominated by the non-dominated solution set and bounded by the nadir point. Unlike other metrics, HV does not require prior knowledge of the true Pareto front and provides a comprehensive assessment of solution quality [45].

Table 2. The parameter setting

Parameter	Meaning	Value
Γ_1	initial temperature	1
Γ_0	final temperature	0.01
I_n	maximum no improvement iteration	20,000
α	cooling rate	0.995
$I_{\max}$	maximum iteration at a temperature	80
Q_1	initial destroy ratio	0.5
Q_0	final destroy ratio	0.05
K	Boltzmann constant	0.025
T	number of maximum generations	4000
Pc	crossover probability	0.9
Pm	mutation probability	0.2
Pop	population size	50

Spacing Metric (θ): The spacing metric evaluates the uniformity of the non-dominated solutions by measuring the standard deviation of the distances between adjacent solutions. A smaller θ value indicates a more evenly distributed Pareto front, which is desirable in multi-objective optimization.

Parameter setting

To ensure fair comparisons, the algorithm parameters for AMOSA, NSGA-II, MOVNS, and IMOVNS are tuned based on initial trials and best practices from related studies. The parameter settings used in this study are presented in Table 2.

All algorithms are implemented using MATLAB R2019a and executed on a personal computer equipped with an Intel Core i7-10700 CPU @ 2.90 GHz, 32 GB of RAM, and an NVIDIA RTX 2080 Super GPU. However, only one CPU core is used in the experiments to maintain computational consistency. The stopping criterion is set at 10,000 objective function evaluations for all algorithms. Each test problem is executed 10 times independently, and the results are averaged to obtain the final performance scores.

Performance Evaluation

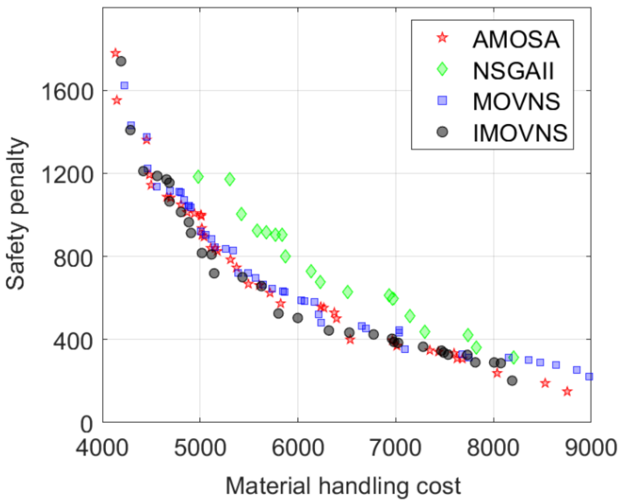
Table 3 presents the comparative performance of IMOVNS, MOVNS, AMOSA, and NSGA-II across all benchmark instances using five evaluation metrics: Euclidean Distance (ED), Spread (SO), Hypervolume (HV), Purity (P), and Spacing Metric (θ). Overall, IMOVNS demonstrates strong and consistent performance, achieving the best average across all metrics. Its advantage is most pronounced in HV and P, where the improvement over other algorithms is both consistent and statistically significant, indicating superior solution diversity and dominance within the Pareto front. In contrast, the improvements in ED and SO are relatively smaller, suggesting that while IMOVNS achieves comparable convergence levels to AMOSA and MOVNS, its main strength lies in maintaining well-distributed and high-quality Pareto sets.

Notably, for more challenging instances such as am13, IMOVNS, AMOSA, and MOVNS achieve similar ED, HV, and P values, but IMOVNS continues to outperform in SO and θ, reflecting its robustness under complex spatial configurations. Conversely, AMOSA and MOVNS show competitive results in a few smaller or less constrained

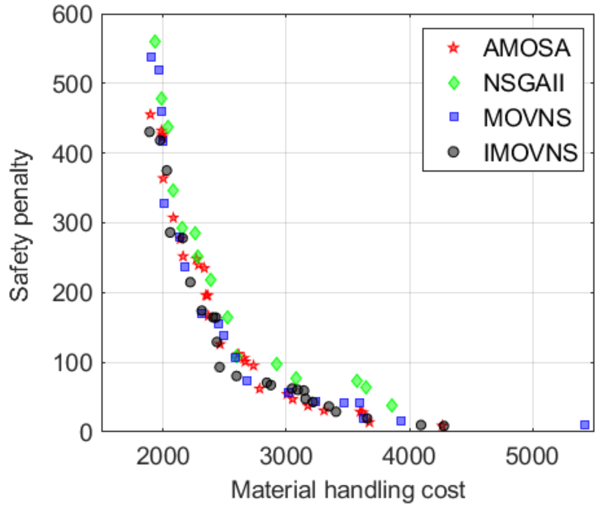
 DOI: 10.25077/josi.v24.n2.p174-197.2025

Table 3. Average indicator values of IMOVNS and the benchmark methods

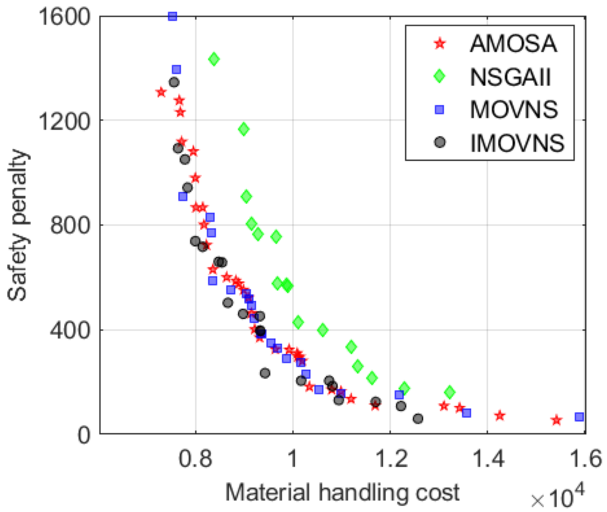
Instance	*ED*				Purity				*SO*				*HV*				***θ***			
	AMOSA	NSGAII	MOVNS	IMOVNS	AMOSA	NSGAII	MOVNS	IMOVNS	AMOSA	NSGAII	MOVNS	IMOVNS	AMOSA	NSGAII	MOVNS	IMOVNS	AMOSA	NSGAII	MOVNS	IMOVNS
S9	0.412	0.470	0.416	0.412	0.276	0.150	0.274	0.300	0.899	0.981	0.909	0.908	0.393	0.328	0.387	0.392	0.020	0.034	0.019	0.020
S9H	0.340	0.416	0.340	0.339	0.264	0.181	0.286	0.270	0.980	1.055	0.992	0.984	0.298	0.242	0.290	0.296	0.029	0.040	0.024	0.032
S10	0.419	0.491	0.417	0.414	0.291	0.167	0.256	0.286	0.821	0.867	0.812	0.816	0.465	0.407	0.460	0.466	0.017	0.023	0.024	0.017
S11	0.435	0.535	0.444	0.440	0.284	0.161	0.259	0.296	0.850	0.912	0.866	0.856	0.431	0.363	0.429	0.434	0.023	0.030	0.024	0.020
am11a	0.270	0.349	0.265	0.258	0.297	0.163	0.264	0.276	0.580	0.625	0.570	0.561	0.574	0.522	0.574	0.575	0.041	0.067	0.054	0.035
am11b	0.344	0.444	0.366	0.350	0.261	0.193	0.266	0.280	0.801	0.867	0.813	0.797	0.442	0.384	0.436	0.446	0.029	0.044	0.031	0.030
am11c	0.425	0.491	0.428	0.423	0.293	0.168	0.255	0.283	0.857	0.916	0.849	0.849	0.433	0.374	0.427	0.434	0.021	0.028	0.028	0.028
am11d	0.304	0.356	0.298	0.297	0.276	0.186	0.249	0.290	0.611	0.667	0.618	0.610	0.551	0.495	0.550	0.551	0.036	0.041	0.042	0.033
am11e	0.307	0.446	0.332	0.326	0.257	0.235	0.243	0.264	0.560	0.630	0.559	0.536	0.594	0.529	0.590	0.598	0.041	0.048	0.037	0.036
am11f	0.302	0.381	0.307	0.307	0.271	0.202	0.261	0.266	0.597	0.649	0.597	0.596	0.582	0.534	0.579	0.584	0.035	0.028	0.035	0.027
am12a	0.308	0.428	0.313	0.303	0.286	0.184	0.250	0.279	0.608	0.693	0.608	0.591	0.608	0.516	0.607	0.609	0.026	0.049	0.027	0.028
am12b	0.299	0.379	0.295	0.288	0.289	0.189	0.254	0.269	0.588	0.681	0.608	0.589	0.581	0.511	0.577	0.579	0.029	0.044	0.042	0.037
am12c	0.361	0.420	0.364	0.361	0.281	0.172	0.256	0.292	0.678	0.733	0.686	0.673	0.548	0.491	0.544	0.552	0.026	0.028	0.029	0.023
am12d	0.378	0.431	0.373	0.365	0.301	0.175	0.261	0.264	0.588	0.624	0.585	0.579	0.594	0.548	0.596	0.598	0.025	0.045	0.032	0.034
am12e	0.280	0.381	0.285	0.252	0.257	0.226	0.230	0.286	0.615	0.691	0.624	0.605	0.500	0.434	0.501	0.503	0.046	0.082	0.061	0.049
am12f	0.325	0.390	0.330	0.326	0.284	0.199	0.247	0.270	0.640	0.684	0.631	0.635	0.567	0.519	0.565	0.568	0.025	0.032	0.033	0.027
am13a	0.332	0.396	0.340	0.333	0.270	0.170	0.264	0.296	0.709	0.748	0.709	0.713	0.518	0.450	0.513	0.518	0.036	0.041	0.034	0.032
am13b	0.302	0.390	0.303	0.300	0.251	0.198	0.259	0.292	0.595	0.679	0.607	0.607	0.590	0.513	0.585	0.587	0.036	0.033	0.036	0.025
am13c	0.261	0.379	0.275	0.265	0.254	0.215	0.289	0.242	0.640	0.719	0.669	0.653	0.449	0.396	0.445	0.447	0.050	0.077	0.063	0.035
am13d	0.360	0.442	0.373	0.353	0.269	0.197	0.258	0.275	0.718	0.786	0.720	0.715	0.474	0.413	0.468	0.474	0.033	0.034	0.037	0.032
am13e	0.261	0.334	0.258	0.258	0.297	0.192	0.256	0.255	0.627	0.681	0.622	0.618	0.536	0.487	0.533	0.539	0.038	0.046	0.047	0.048
am13f	0.269	0.379	0.278	0.269	0.264	0.202	0.258	0.277	0.645	0.737	0.653	0.654	0.498	0.436	0.495	0.498	0.040	0.045	0.042	0.036
14a	0.304	0.419	0.309	0.307	0.279	0.163	0.269	0.289	0.646	0.734	0.649	0.653	0.545	0.455	0.540	0.545	0.040	0.037	0.041	0.034
14b	0.352	0.426	0.369	0.361	0.279	0.191	0.253	0.276	0.732	0.805	0.741	0.746	0.490	0.436	0.488	0.492	0.031	0.029	0.024	0.023
P15	0.378	0.470	0.377	0.368	0.288	0.181	0.256	0.276	0.689	0.758	0.679	0.683	0.539	0.454	0.539	0.539	0.032	0.041	0.040	0.052
P17	0.404	0.482	0.397	0.391	0.288	0.170	0.259	0.284	0.719	0.773	0.721	0.718	0.513	0.438	0.504	0.513	0.028	0.041	0.032	0.029
P18	0.320	0.403	0.318	0.312	0.280	0.186	0.244	0.289	0.710	0.782	0.710	0.705	0.505	0.431	0.501	0.504	0.039	0.044	0.038	0.046
Average	0.335	0.420	0.340	0.332	0.277	0.186	0.258	0.279	0.693	0.758	0.697	0.691	0.512	0.448	0.508	0.512	0.032	0.042	0.036	0.032



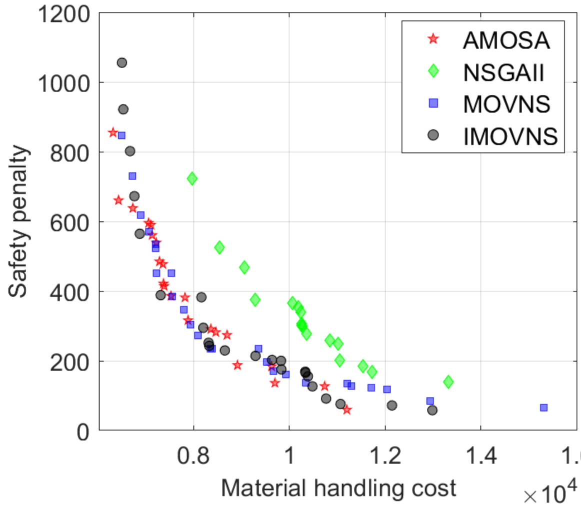
a) am11c, $n = 11$



b) am12b, $n = 12$



c) am13d, $n = 13$



d) P18, $n = 18$

Figure 9. Pareto front obtained by each method in four instances with $n = \{11,12,13,18\}$

cases, demonstrating their capability in local refinement but with less consistency in global coverage. These observations indicate that IMOVNS achieves a balanced trade-off between convergence and diversity, making it a reliable approach for solving safety-aware and geometry-constrained DRLP instances.

In contrast, NSGA-II shows the weakest performance in nearly all aspects, especially in terms of convergence and diversity. Although MOVNS and AMOSA generate relatively competitive results, they remain less robust than IMOVNS across different problem instances. The Spacing Metric (SM) values further confirm that IMOVNS produces more uniformly distributed solutions compared to the benchmarks.

Figure 9 shows how many solutions from each algorithm contribute to the global Pareto front across all tested instances. The results clearly indicate that IMOVNS provides the highest proportion of non-dominated solutions, highlighting its strong ability to produce trade-offs that outperform those from competing methods. Although AMOSA and MOVNS also offer meaningful contributions, their shares are noticeably smaller. NSGA-II has only a minor presence on the Pareto front, suggesting that many of its solutions are dominated by those generated through

 DOI: 10.25077/josi.v24.n2.p174-197.2025

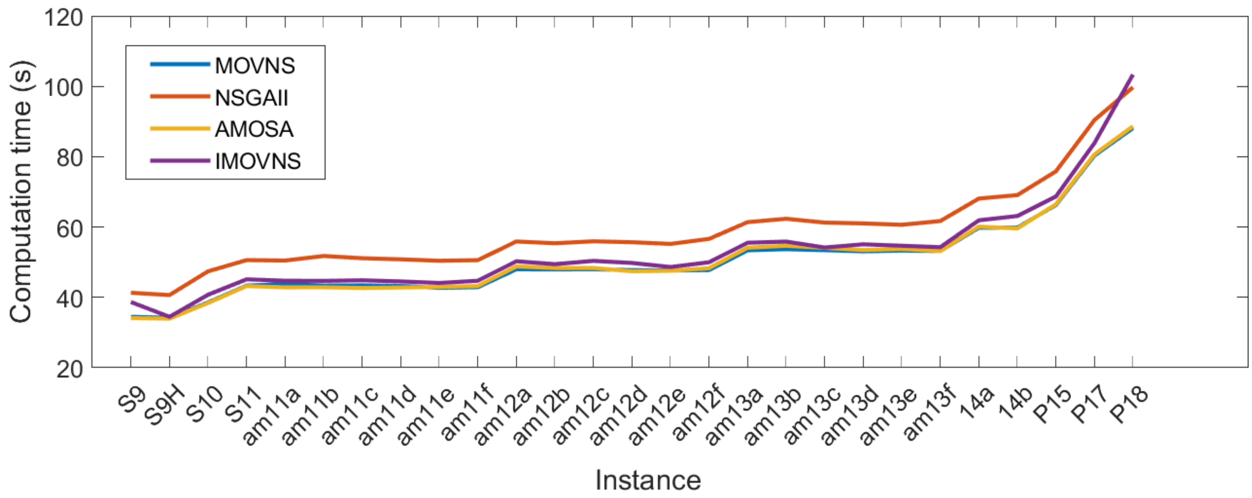


Figure 10. Computation time

the other approaches. This graphical evidence supports the numerical results reported in Table 3, confirming that IMOVNS not only performs well in terms of convergence and diversity, but also produces solutions that are more relevant in practical decision-making.

The consistently superior performance of IMOVNS demonstrates that the methodological improvements introduced in the framework, particularly the adaptive archive mechanism and the acceptance rules inspired by AMOSA, substantially enhance its optimization capability when addressing DRLP instances with complex constraints.

Computation time Evaluation

Although all algorithms were executed under the same stopping criterion of 10,000 objective function evaluations, their actual computational times differ significantly due to variations in their internal structures and operational complexities. Among the four algorithms, NSGA-II incurred the highest average runtime. This is primarily attributed to its population-based mechanism, which involves complex procedures such as selection, crossover, and mutation across multiple individuals in each generation. These operations inherently demand more computational resources and time.

In contrast, IMOVNS, AMOSA, and MOVNS operate as trajectory-based, single-solution metaheuristics that progress through local neighborhood search and solution acceptance rules. Such approaches typically offer lower computational effort per iteration. Nonetheless, IMOVNS incurs slightly higher computing time compared with AMOSA and MOVNS due to its adaptive archive mechanism and probabilistic acceptance procedure derived from AMOSA. Although these elements introduce a modest runtime overhead, they contribute meaningfully to improving the overall solution quality.

Figure 10 presents the average computation time of each method across all benchmark instances. The results indicate that IMOVNS provides a reasonable compromise between runtime and solution performance, making it a practical option. Although AMOSA and MOVNS are faster on average, the improvement in optimization outcomes obtained through IMOVNS justifies its somewhat higher execution time.

In real industrial applications, particularly those where solution quality takes precedence over minimal processing time, such as layout design involving safety restrictions, the proposed IMOVNS remains both effective and suitable for deployment.

Table 4. Paired sample t-test between the IMOVNS and the benchmark methods

Criteria (metric)	Benchmark methods					
	AMOSA		NSGAII		MOVNS	
	p-value	description	p-value	description	p-value	description
ED	0.8439	Fail to reject H_0	7.3×10^{-8}	H_0 is rejected; IMOVNS performs better	0.6164	Fail to reject H_0
P	0.7263	Fail to reject H_0	5.3×10^{-24}	H_0 is rejected; IMOVNS performs better	3.4×10^{-7}	H_0 is rejected; IMOVNS performs better
SO	0.9479	Fail to reject H_0	0.0312	H_0 is rejected; IMOVNS performs better	0.8491	Fail to reject H_0
HV	0.9693	Fail to reject H_0	0.0019	H_0 is rejected; IMOVNS performs better	0.8333	Fail to reject H_0
θ	0.9262	Fail to reject H_0	0.0035	H_0 is rejected; IMOVNS performs better	0.1423	Fail to reject H_0
T	0.5109	Fail to reject H_0	0.1259	Fail to reject H_0	0.5339	Fail to reject H_0

Statistical analysis

To validate the statistical significance of the observed differences in performance among the four algorithms, a series of paired-sample t-tests were conducted using a significance level of α = 0.05. The tests were applied to the results obtained from ten independent runs on 27 problem instances, using the five multi-objective performance indicators described previously: Euclidean Distance (ED), Purity (P), Sum of Objectives (SO), Hypervolume (HV), and Spacing Metric (SM).

Table 4 summarizes the results of the t-tests between IMOVNS and each of the benchmark algorithms (NSGA-II, AMOSA, and MOVNS). The results of the t-tests indicate that IMOVNS achieves statistically significant improvements over NSGA-II across all evaluation metrics, confirming its advantage in convergence and solution quality. However, against AMOSA and MOVNS, the differences are generally not statistically significant, suggesting that these methods perform comparably on several metrics.

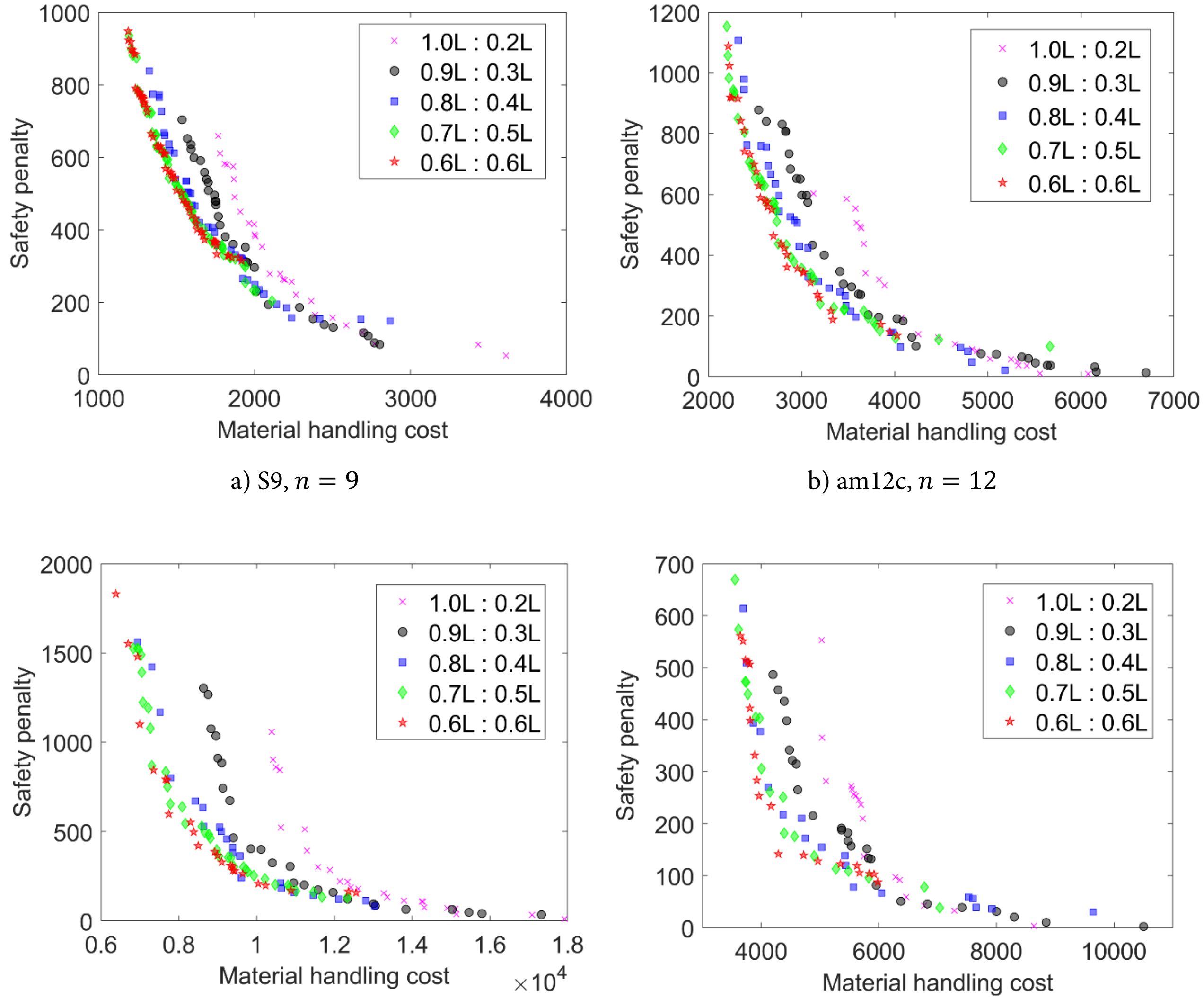
A key advantage of IMOVNS is its steady performance across different problem sets and evaluation criteria, indicating stronger robustness rather than universal dominance in every scenario. Accordingly, the interpretation of the findings has been refined to emphasize that IMOVNS consistently outperforms NSGA-II based on statistical evidence and remains highly competitive with both AMOSA and MOVNS. Although AMOSA and MOVNS can produce strong results, their outcomes tend to be less stable and generally weaker than those of IMOVNS in terms of convergence behaviour and uniformity across diverse layouts. In addition, the relatively small dispersion in IMOVNS results, as demonstrated by the low standard deviations over multiple independent executions, further supports its reliability. This highlights a central benefit of the IMOVNS framework: its ability to deliver consistent solution quality even when facing problem variability and random initial solution states.

Overall, the statistical evaluation clearly demonstrates the superiority of IMOVNS, especially relative to NSGA-II, and validates its capability to produce high-quality Pareto solutions for multi-objective DRLP problems that involve both safety and geometric constraints.

Managerial insights

Beyond algorithmic performance, the findings from this study also offer valuable managerial insights that can support decision-making in facility layout planning, particularly in manufacturing environments where safety, space constraints, and material handling efficiency must be jointly considered.

 DOI: 10.25077/josi.v24.n2.p174-197.2025



a) S9, $n = 9$

b) am12c, $n = 12$

c) am13d, $n = 13$

d) P15, $n = 15$

Figure 11. Illustrative Pareto fronts showing the trade-off between material handling cost and safety penalty under different row length constraints.

Insight 1: Trade-off Between Row Length Balance and Objective Priorities

A sensitivity analysis was conducted by varying the imbalance between the two row lengths across different test instances. The results reveal that as the imbalance grows (e.g., from 0.6 : 0.6L to 1.0 : 0.2L), the solution set shifts, achieving lower safety penalties (y-axis) but at the expense of higher material handling costs (x-axis). When one row becomes substantially longer than the other, machine placement options increase, making it easier to satisfy safety distance requirements and thereby reducing penalty values. However, this added flexibility typically comes with higher material handling costs, as greater spacing between machines leads to longer transportation distances.

In contrast, layouts with more balanced row lengths tend to minimize handling costs but are more likely to breach safety distance constraints, as machines are packed more densely within restricted floor space. This demonstrates an inherent trade-off between efficient space utilization and adherence to safety requirements.

Figure 11 illustrates this relationship, showing how varying levels of row imbalance influence the two optimization objectives. The results reveal a clear pattern: increasing row imbalance improves safety compliance but leads to higher transportation cost. This outcome highlights that greater spatial flexibility can alleviate safety restrictions, although it reduces routing efficiency.

Insight 2: Flexibility Through Pareto-optimal Solutions

A second important observation relates to how the Pareto front generated by IMOVNS can be interpreted in a practical planning context. Instead of producing a generic collection of trade-offs, the obtained Pareto front highlights distinct layout patterns that emerge when safety distances and row geometry are jointly considered. For instance, one Pareto-optimal configuration that minimizes material handling cost tends to have tighter machine spacing and shorter transport paths, which increases safety penalties. This design is well suited to operations where throughput and routing efficiency are more critical than spacing—such as automated production lines or facilities that rely heavily on AGV systems. On the other hand, a Pareto-optimal solution that prioritizes minimum safety penalty features wider spacing between machines, especially within the shorter row, reducing congestion and ensuring safer separation. However, this arrangement also lengthens transport distances, resulting in higher material handling cost.

These contrasting patterns demonstrate how safety and geometric constraints jointly shape the Pareto front and provide valuable managerial flexibility. Facility designers can choose layout options based on their operational needs, preferring compact, cost-efficient layouts for highly automated settings, or more spacious, safety-oriented layouts for environments involving human operators or hazardous materials. This finding emphasizes that the Pareto front produced by IMOVNS does more than improve optimization metrics. It converts computational results into practical decision choices that balance operational efficiency with safety compliance.

CONCLUSION

This study extends the Double Row Layout Problem by incorporating two practical constraints—minimum safety spacing and geometric row length limitations—into a bi-objective MINLP model aimed at minimizing material handling costs and safety penalties. To address its complexity, a novel metaheuristic, IMOVNS, was introduced, enhancing the original MOVNS with adaptive archiving and probabilistic acceptance to improve convergence and diversity in Pareto front generation. IMOVNS was evaluated on 27 benchmark DRLP instances and compared with NSGA-II, AMOSA, and MOVNS using five widely used multi-objective performance indicators. Results show that IMOVNS consistently outperforms NSGA-II and performs comparably or better than AMOSA and MOVNS, achieving closer convergence to the ideal Pareto front, greater diversity, and more uniform trade-off distribution. Statistical tests confirm its superiority, particularly over NSGA-II, and runtime analysis indicates that IMOVNS is computationally feasible for industrial-scale applications. The analysis reveals a trade-off between row length symmetry and optimization outcomes: layouts with uneven row lengths generally improve safety compliance but increase handling effort. The Pareto solutions generated by IMOVNS provide practitioners with diverse layout options, enabling them to balance safety, operational cost, and space utilization according to specific priorities. This study is limited by the non-linear nature of the proposed model, which increases computational complexity, and by the parameter settings used for the metaheuristics, which may affect generalizability. Future research should focus on developing linear or hybrid MILP formulations to improve efficiency, testing the scalability of IMOVNS on larger and more complex layouts, and validating the approach with real industrial data to confirm its practical applicability.

ACKNOWLEDGEMENT

The authors would like to express their sincere gratitude to all individuals and institutions who contributed to the completion of this manuscript. We extend our appreciation to the journal's editorial team and reviewers for their valuable time, constructive feedback, and insightful comments, which greatly improved the quality and clarity of this work.

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] C. Guan, Z. Zhang, S. Liu, and J. Gong, "Multi-objective particle swarm optimization for multi-workshop facility layout problem," J. Manuf. Syst., vol. 53, pp. 32–48, Oct. 2019, doi: 10.1016/j.jmsy.2019.09.004.

[2] A. Mohamadghasemi and A. Hadi-Vencheh, "An integrated synthetic value of fuzzy judgments and nonlinear programming methodology for ranking the facility layout patterns," Comput. Ind. Eng., vol. 62, no. 1, pp. 342–348, Feb. 2012, doi: 10.1016/j.cie.2011.10.004.

[3] J. Liu, H. Zhang, K. He, and S. Jiang, "Multi-objective particle swarm optimization algorithm based on objective space division for the unequal-area facility layout problem," Expert Syst. Appl., vol. 102, pp. 179–192, Jul. 2018, doi: 10.1016/j.eswa.2018.02.035.

[4] B. Keller and U. Buscher, "Single row layout models," Eur. J. Oper. Res., vol. 245, no. 3, pp. 629–644, Sep. 2015, doi: 10.1016/j.ejor.2015.03.016.

[5] J. Chung and J. M. A. Tanchoco, "The double row layout problem," Int. J. Prod. Res., vol. 48, no. 3, pp. 709–727, Feb. 2010, doi: 10.1080/00207540802192126.

[6] G. Aiello, G. La Scalia, and M. Enea, "A multi objective genetic algorithm for the facility layout problem based upon slicing structure encoding," Expert Syst. Appl., vol. 39, no. 12, pp. 10352–10358, Sep. 2012, doi: 10.1016/j.eswa.2012.01.125.

[7] S. Ingole and D. Singh, "Unequal-area, fixed-shape facility layout problems using the firefly algorithm," Eng. Optim., vol. 49, no. 7, pp. 1097–1115, Jul. 2017, doi: 10.1080/0305215X.2016.1235327.

[8] Y. H. Lee and M. H. Lee, "A shape-based block layout approach to facility layout problems using hybrid genetic algorithm," Comput. Ind. Eng., vol. 42, no. 2–4, pp. 237–248, Apr. 2002, doi: 10.1016/S0360-8352(02)00018-9.

[9] J. E. C. Arroyo, R. Dos Santos Ottoni, and A. De Paiva Oliveira, "Multi-objective Variable Neighborhood Search Algorithms for a Single Machine Scheduling Problem with Distinct due Windows," Electron. Notes Theor. Comput. Sci., vol. 281, pp. 5–19, Dec. 2011, doi: 10.1016/j.entcs.2011.11.022.

[10] A. Duarte, J. J. Pantrigo, E. G. Pardo, and N. Mladenovic, "Multi-objective variable neighborhood search: an application to combinatorial optimization problems," J. Glob. Optim., vol. 63, no. 3, pp. 515–536, Nov. 2015, doi: 10.1007/s10898-014-0213-z.

[11] P. J. M. Van Laarhoven and E. H. L. Aarts, "Simulated annealing," in Simulated Annealing: Theory and Applications, Dordrecht: Springer Netherlands, 1987, pp. 7–15. doi: 10.1007/978-94-015-7744-1_2.

[12] Z. Zhang and C. C. Murray, "A corrected formulation for the double row layout problem," Int. J. Prod. Res., vol. 50, no. 15, pp. 4220–4223, Aug. 2012, doi: 10.1080/00207543.2011.603371.

[13] C. C. Murray, A. E. Smith, and Z. Zhang, "An efficient local search heuristic for the double row layout problem with asymmetric material flow," Int. J. Prod. Res., vol. 51, no. 20, pp. 6129–6139, Oct. 2013, doi: 10.1080/00207543.2013.803168.

[14] A. R. S. Amaral, "A mixed-integer programming formulation for the double row layout of machines in manufacturing systems," Int. J. Prod. Res., vol. 57, no. 1, pp. 34–47, Jan. 2019, doi: 10.1080/00207543.2018.1457811.

[15] A. Drira, H. Pierreval, and S. Hajri-Gabouj, "Facility layout problems: A survey," Annu. Rev. Control, vol. 31, no. 2, pp. 255–267, Jan. 2007, doi: 10.1016/j.arcontrol.2007.04.001.

[16] S. P. Singh and R. R. K. Sharma, "A review of different approaches to the facility layout problems," Int. J. Adv. Manuf. Technol., vol. 30, no. 5–6, pp. 425–433, Sep. 2006, doi: 10.1007/s00170-005-0087-9.

[17] J. Chae and A. C. Regan, "A mixed integer programming model for a double row layout problem," Comput. Ind. Eng., vol. 140, p. 106244, Feb. 2020, doi: 10.1016/j.cie.2019.106244.

[18] S. Duan and L. Kang, "An Enhanced Multiobjective Double Row Layout Model considering the Machine Breakdowns," Comput. Intell. Neurosci., vol. 2022, pp. 1–14, Aug. 2022, doi: 10.1155/2022/6289609.

[19] Z. Zhang, L. Wu, Z. Wu, W. Zhang, S. Jia, and T. Peng, "Energy-Saving Oriented Manufacturing Workshop Facility Layout: A Solution Approach Using Multi-Objective Particle Swarm Optimization," Sustainability, vol. 14, no. 5, p. 2788, Feb. 2022, doi: 10.3390/su14052788.

[20] M. Gülşen, C. C. Murray, and A. E. Smith, "Double-row facility layout with replicate machines and split flows," Comput. Oper. Res., vol. 108, pp. 20–32, Aug. 2019, doi: 10.1016/j.cor.2019.03.009.

[21] P. Pérez-Gosende, J. Mula, and M. Díaz-Madroñero, "Facility layout planning. An extended literature review," Int. J. Prod. Res., vol. 59, no. 12, pp. 3777–3816, Jun. 2021, doi: 10.1080/00207543.2021.1897176.

[22] A. P. Rifai, S. T. Windras Mara, H. Ridho, and R. Norcahyo, "The double row layout problem with safety consideration: a two-stage variable neighborhood search approach," J. Ind. Prod. Eng., vol. 39, no. 3, pp. 181–195, Apr. 2022, doi: 10.1080/21681015.2021.1968964.

[23] W. Isnaini, A. P. Rifai, N. M. E. Nurmasari, N. A. Masruroh, I. B. Dharma, and V. E. Andriani, "Sequential use of blocplan, solver, and particle swarm optimization (PSO) to optimize the double row facility layout," Int. J. Prod. Manag. Eng., vol. 12, no. 2, pp. 117–124, May 2024, doi: 10.4995/ijpme.2024.20061.

[24] X. Zuo, C. C. Murray, and A. E. Smith, "Solving an Extended Double Row Layout Problem Using Multiobjective Tabu Search and Linear Programming," IEEE Trans. Autom. Sci. Eng., vol. 11, no. 4, pp. 1122–1132, Oct. 2014, doi: 10.1109/TASE.2014.2304471.

[25] S. Wang, X. Zuo, X. Liu, X. Zhao, and J. Li, "Solving dynamic double row layout problem via combining simulated annealing and mathematical programming," Appl. Soft Comput., vol. 37, pp. 303–310, Dec. 2015, doi: 10.1016/j.asoc.2015.08.023.

[26] F. Jolai, R. Tavakkoli-Moghaddam, and M. Taghipour, "A multi-objective particle swarm optimisation algorithm for unequal sized dynamic facility layout problem with pickup/drop-off locations," Int. J. Prod. Res., vol. 50, no. 15, pp. 4279–4293, Aug. 2012, doi: 10.1080/00207543.2011.613863.

[27] H. Rezazadeh, M. Ghazanfari, M. Saidi-Mehrabad, and S. Jafar Sadjadi, "An extended discrete particle swarm optimization algorithm for the dynamic facility layout problem," J. Zhejiang Univ.-Sci. A, vol. 10, no. 4, pp. 520–529, Apr. 2009, doi: 10.1631/jzus.A0820284.

[28] A. Derakhshan Asl, K. Y. Wong, and M. K. Tiwari, "Unequal-area stochastic facility layout problems: solutions using improved covariance matrix adaptation evolution strategy, particle swarm optimisation, and genetic algorithm," Int. J. Prod. Res., vol. 54, no. 3, pp. 799–823, Feb. 2016, doi: 10.1080/00207543.2015.1070217.

[29] S. RazaviAlavi and S. AbouRizk, "Genetic Algorithm–Simulation Framework for Decision Making in Construction Site Layout Planning," J. Constr. Eng. Manag., vol. 143, no. 1, p. 04016084, Jan. 2017, doi: 10.1061/(ASCE)CO.1943-7862.0001213.
[30] N. Mladenović and P. Hansen, "Variable neighborhood search," Comput. Oper. Res., vol. 24, no. 11, pp. 1097–1100, Nov. 1997, doi: 10.1016/S0305-0548(97)00031-2.
[31] D. Matić, J. Kratica, V. Filipović, and D. Dugošija, "Variable neighborhood search for Multiple Level Warehouse Layout Problem," Electron. Notes Discrete Math., vol. 39, pp. 161–168, Dec. 2012, doi: 10.1016/j.endm.2012.10.022.
[32] K. S. N. Ripon, K. Glette, K. N. Khan, M. Hovin, and J. Torresen, "Adaptive variable neighborhood search for solving multi-objective facility layout problems with unequal area facilities," Swarm Evol. Comput., vol. 8, pp. 1–12, Feb. 2013, doi: 10.1016/j.swevo.2012.07.003.
[33] P. Hansen and N. Mladenović, "Variable Neighborhood Search," in Search Methodologies, E. K. Burke and G. Kendall, Eds., Boston, MA: Springer US, 2005, pp. 211–238. doi: 10.1007/0-387-28356-0_8.
[34] B. Jarboui, H. Derbel, S. Hanafi, and N. Mladenović, "Variable neighborhood search for location routing," Comput. Oper. Res., vol. 40, no. 1, pp. 47–57, Jan. 2013, doi: 10.1016/j.cor.2012.05.009.
[35] S. Bandyopadhyay, S. Saha, U. Maulik, and K. Deb, "A Simulated Annealing-Based Multiobjective Optimization Algorithm: AMOSA," IEEE Trans. Evol. Comput., vol. 12, no. 3, pp. 269–283, Jun. 2008, doi: 10.1109/TEVC.2007.900837.
[36] L. F. Muller, S. Spoorendonk, and D. Pisinger, "A hybrid adaptive large neighborhood search heuristic for lot-sizing with setup times," Eur. J. Oper. Res., vol. 218, no. 3, pp. 614–623, May 2012, doi: 10.1016/j.ejor.2011.11.036.
[37] D. Freitas, L. G. Lopes, and F. Morgado-Dias, "Particle Swarm Optimisation: A Historical Review Up to the Current Developments," Entropy, vol. 22, no. 3, p. 362, Mar. 2020, doi: 10.3390/e22030362.
[38] D. M. Simmons, "One-Dimensional Space Allocation: An Ordering Algorithm," Oper. Res., vol. 17, no. 5, pp. 812–826, Oct. 1969, doi: 10.1287/opre.17.5.812.
[39] A. R. S. Amaral, "On the exact solution of a facility layout problem," Eur. J. Oper. Res., vol. 173, no. 2, pp. 508–518, Sep. 2006, doi: 10.1016/j.ejor.2004.12.021.
[40] A. R. S. Amaral, "An Exact Approach to the One-Dimensional Facility Layout Problem," Oper. Res., vol. 56, no. 4, pp. 1026–1033, Aug. 2008, doi: 10.1287/opre.1080.0548.
[41] A. R. S. Amaral, "A new lower bound for the single row facility layout problem," Discrete Appl. Math., vol. 157, no. 1, pp. 183–190, Jan. 2009, doi: 10.1016/j.dam.2008.06.002.
[42] L. D. Secchin and A. R. S. Amaral, "An improved mixed-integer programming model for the double row layout of facilities," Optim. Lett., vol. 13, no. 1, pp. 193–199, Feb. 2019, doi: 10.1007/s11590-018-1263-9.
[43] M. Laszczyk and P. B. Myszkowski, "Survey of quality measures for multi-objective optimization: Construction of complementary set of multi-objective quality measures," Swarm Evol. Comput., vol. 48, pp. 109–133, Aug. 2019, doi: 10.1016/j.swevo.2019.04.001.
[44] M. Li and X. Yao, "Quality Evaluation of Solution Sets in Multiobjective Optimisation: A Survey," ACM Comput. Surv., vol. 52, no. 2, pp. 1–38, Mar. 2020, doi: 10.1145/3300148.
[45] S. Rostami and F. Neri, "A fast hypervolume driven selection mechanism for many-objective optimisation problems," Swarm Evol. Comput., vol. 34, pp. 50–67, Jun. 2017, doi: 10.1016/j.swevo.2016.12.002.

AUTHORS BIOGRAHY

Achmad Pratama Rifai received his B.E. degree from Gadjah Mada University, Yogyakarta, Indonesia, in 2011 and his M. Eng degree in Manufacturing Engineering from University of Malaya, Malaysia, in 2014, and his Ph.D. degree, in Integrated Design Engineering, from Keio University Japan, in 2019. Since 2020, he has been a lecturer at Gadjah Mada University. His research interests include Manufacturing and production system, Optimization, Metaheuristic, Machine vision, Machining inspection.

Setyo Tri Windras Mara is a researcher with in-depth knowledge in mathematical modelling, supply chain optimization, and computational intelligence methods. I am currently doing my PhD study in Systems Engineering at University of New South Wales. I received my Master degrees from Taiwan Tech (Industrial Management) and Universitas Gadjah Mada (Industrial Engineering). I have huge interests in domains of well-being, such as sustainable logistics and healthcare operations management.

Rachmadi Norcahyo is junior lecturer at Department of Mechanical and Industrial Engineering, Universitas Gadjah Mada, Indonesia. Previously working as a research assistant at Institut Teknologi Sepuluh Nopember, Surabaya Indonesia. On working project: drilling and milling process optimization on carbon fiber and glass fiber material using statistical and soft computing method.

Andri Nasution received his B.E. degree from the University of Sumatra Utara, Medan, Indonesia, in 2009 and his Master of Engineering degree in Technology Management from the University of Sumatra Utara, Medan, Indonesia, in 2015. Since 2019, he has been a lecturer at the University of North Sumatra. His research interests include Manufacturing and production system, Optimization, Metaheuristic and 3D Printing.

 DOI: 10.25077/josi.v24.n2.p174-197.2025