Received 22 December 2022, accepted 8 January 2023, date of publication 13 January 2023, date of current version 1 February 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3236843

RESEARCH ARTICLE

Effective Scheduling of Multi-Load Automated Guided Vehicle in Spinning Mill: A Case Study

PARKAVI KRISHNAMOORTHY[1], **N. SATHEESH**[2], **D. SUDHA**[3], **SUDHAKAR SENGAN**[4], **(Member, IEEE)**, **MESHAL ALHARBI**[5], **DENIS A. PUSTOKHIN**[6], **IRINA V. PUSTOKHINA**[7], **AND ROY SETIAWAN**[8]

[1]School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, Tamil Nadu 600127, India
[2]Department of Computer Science and Engineering, St. Martin's Engineering College, Secunderabad, Telangana 500100, India
[3]Department of Computer Science and Engineering, Panimalar Engineering College, Chennai, Tamil Nadu 600123, India
[4]Department of Computer Science and Engineering, PSN College of Engineering and Technology, Tirunelveli, Tamil Nadu 627152, India
[5]Department of Computer Science, College of Computer Engineering and Sciences, Prince Sattam Bin Abdulaziz University, Al-Kharj 16278, Saudi Arabia
[6]Department of Logistics, State University of Management, 109542 Moscow, Russia
[7]Department of Entrepreneurship and Logistics, Plekhanov Russian University of Economics, 117997 Moscow, Russia
[8]Department Management, Universitas Kristen Petra, Surabaya 60236, Indonesia

Corresponding author: Parkavi Krishnamoorthy (parkavi.k@vit.ac.in)

ABSTRACT In the Flexible Manufacturing System (FMS), where material processing is carried out in the form of tasks from one department to another, the use of Automated Guided Vehicles (AGVs) is significant. The application of multiple-load AGVs can be understood to boost FMS throughput by multiple orders of magnitude. For the transportation of materials and items inside a warehouse or manufacturing plant, an AGV, a mobile robot, offers extraordinary industrial capabilities. The technique of allocating AGVs to tasks while taking into account the cost and time of operations is known as AGV scheduling. Most research has exclusively addressed single-objective optimization, whereas multi-objective scheduling of AGVs is a complex combinatorial process without a single solution, in contrast to single-objective scheduling. This paper presents the integrated Local Search Probability-based Memetic Water Cycle (LSPM-WC) algorithm using a spinning mill as a case study. The scheduling model's goal is to maximize machine efficiency. The scheduling of the statistical tests demonstrated the applicability of the proposed model in lowering the makespan and fitness values. The mean AGV operating efficiency was higher than the other estimated models, and the LSPM-WC surpassed the different algorithms to produce the best result.

INDEX TERMS Manufacturing system, automated guided vehicles, computer integrated manufacturing, water cycle algorithm, makespan, spinning mill.

I. INTRODUCTION

Industrial Revolution 4.0 technologies have increased Manufacturing System (MS) adaptability. Some emerging technologies that make MS more versatile are the IoT, big data, AI, additive manufacturing, cutting-edge robotics, VR, cloud computing, and simulation. The workstations in the Flexible Manufacturing System (FMS) is rearranged for different operations and procedures to its modular architecture [1]. As is evident, FMS are excellent venues for Industrial Revolution 4.0 research since they have features like networked interconnection, data collection, and distributed intelligence [2].

In the manufacturing industry, material handling entails transporting any material or load inside the confines of the shop floor, or at the most, to warehouses or vehicles. The most recent development in material handling is the use of Automated Guided Vehicles (AGVs). AGVs seem to be portable robots that navigate along a predetermined path. It can either store the material upon these overhead beds during conveyance or carry the material stored on them using trolleys and trailers affixed rearward to them. They transport loads inside warehouses or manufacturing facilities where there is a need for continuous material movement with minimal or no

The associate editor coordinating the review of this manuscript and approving it for publication was Zhiwu Li.

FIGURE 1. Flow chart of spinning with the different processes.

human interaction. Today, they are used in almost all industries, like the food, pharmaceutical, automobile, and textile sectors, handling anything from raw materials to finished goods.

The use of AGVs for all material handling requirements across all industries is increasing daily. The industry has benefited greatly from it. Some of these are listed below.

(i) Consider a single robot moving a weight of 1000 *kg* versus a labourer performing the same task manually. AGVs are unquestionably superior to manual handling. They not only save time, but they also save much human effort.
(ii) Comparatively, AGVs are much simpler to install than conveyor belts. Aside from installing the initial guidance path, they don't need any machinery or maintenance. Unlike conveyor belts, they do not permanently block large spaces. If it is not used, it is kept away in a small area, allowing its movement throughout the entire shop floor.
(iii) To become a skilled forklift operator, a person must undergo months of instruction. Due to the absence of human interference, using the AGVs is simple from the first day of installation.

AGVs' safety, which is crucial in every industry, is their main benefit over other alternatives. Some robots, which have intelligent safety features and sensors, ensure no damage is done to the machinery, products, or workers, fostering a better work environment. The AGVs' only drawback is their high capital cost, which makes businesses reluctant to choose them.

India is one of the world's top garment and textile producers. The Indian domestic textile and garment industry accounts for 7% of the total industry output and 2% of the country's GDP. In 2020–2021, the contribution of apparel, textile, and handicraft exports made up 11.4% of all Indian exports. India accounts for 4% of the global textile and apparel trade. India is the 6th largest exporter of apparel and textiles worldwide. From fibers, yarns, and fabrics to clothing, India's apparel and textile industry has strengths along the whole value chain. The traditional handloom, handicrafts, wool, silk products, and organized textile industry in India make up a large portion of the wide-ranging Indian textile industry. The automated spinning, weaving, processing, and textile manufacturing processes used in India's structured textile industry are capital-intensive technologies that distinguish them from other textile-manufacturing countries. This paper focuses on the spinning mill industry. Because clothing is one of the most basic human needs, and spinning is one of the world's first industries, it will undoubtedly continue to boom. The following procedures shown in Figure 1 are included when this work is spinning.

A. PROCESS DESCRIPTION OF SPINNING

1) BALE MANAGEMENT

The cotton plant is used for fiber collection. All fiber is treated to eliminate massive dust, such as plant leaves, soil, and other contaminants. Finally, the fibre bale's weight (218 to 225 *kg*) and sizes (1.400 x 0.53 x 0.69 *m*) are determined. After ginning, the bale typically consists of compressed lint that has been secured with wire.

2) BLOW ROOM

The second spinning phase is the blow room, where the bales are opened and arranged one at a time for fiber mixing according to the information on each bale. Information such as color grade, weight (218 *kg* to 225 *kg*), size (1.400 *m* X 0.53 *m* X 0.69 *m*), and batch number are all recorded in the bale. The objective of this process is to make the fiber lap.

3) CARDING MACHINE

The lap is fed over the carding machine to make a carded sliver. This procedure aims to straighten the fiber and cut out any short threads.

4) DRAW FRAME

A draw frame is a machine that produces one drawn sliver by drafting two or more carded slivers between three drafting rollers. Of these three rollers, the third is the fastest, followed by the second, which is the shortest of the three. A second drawing frame is also used if more drafting is required.

5) SIMPLEX/SPEED FRAME

For producing roving yarn with a greater diameter than finished yarn, the drawn sliver is treated in the simplex/speed frame. The roving yarn has relatively little strength and is factory-made without a twist.

6) RING FRAME

The roving yarn is twisted on a ring frame machine to produce the finished yarn. This procedure gives the yarn a good quantity of turning strength.

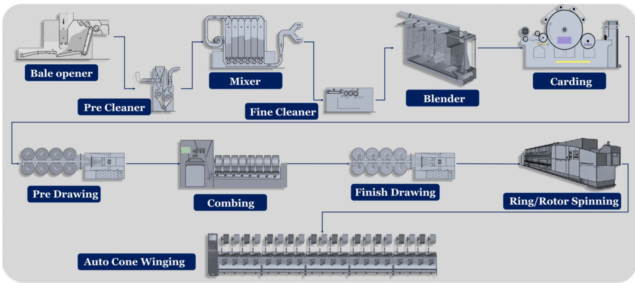


FIGURE 2. Spinning Mill process diagram.

7) AUTO CONER

Generally, an auto coner is mainly used to make cone packages of different dimensions. After that, the yarn is packaged for use and delivery.

Through Computer Integrated Manufacturing (CIM) in each production stage, automation in the spinning mill industry is possible. It is possible to monitor/control nearly all procedures for yarn production, from opening and blending to spinning, winding, and twisting, using CIM systems, as illustrated in Figure 2. Applications range widely, including mill management, budgeting, order tracking, maintenance, and inventory control. Most companies today provide advanced controls for CIM-compatible opening, blending, carding, and other fibre preparation equipment. There are ring-spinning machines with separate spindle drives that are highly flexible and easily adaptable to the CIM concept. On-machine electronics connected to a computer network can regulate silver weights and modify the levels. In online quality control for carding and drawing, the frequency analysis of the defects is used to perform spectral and identify the problem's root cause. Because robotic machines perform automatic end piecing and doffing, a giant spinning mill is run by a limited number of employees.

Every procedure depends heavily on other departments. Therefore, a single plant is spread over an area with designated spaces separating each activity. In any case, transporting such material from one location to another is tedious because it is bulky. Imagine transporting tonnes of cotton bales from a warehouse toward a spinning mill or carded yarn out of a carding department to a drawing frame. Transporting the material for textiles still requires manual labour in so many industries. However, it becomes nearly impossible to complete these tasks manually when the industries develop. Pallet trucks, forklifts, and overhead pick-and-place machines emerged, and many textile industries still use these types of machinery today. But men still must operate each of these. Additionally, forklifts and pallet trucks are the primary causes behind shop floor accidents that Jeopardise workers' safety at the point of risk.

The AGVs enter this situation because they are 100% autonomous and don't require human interaction. The possibility of errors is significantly minimised as a result. They require much less maintenance than traditional forklifts and pallet trucks. It guarantees fully automatic load carriage between one department and another, including automated bale, cart, or trolley picking and placing. Yarns and bobbins are transported using the AGV bed's unique mechanism without hooking or tying. It enables 100% automation by reducing manual time and effort. AGVs can operate without issue in such a hot and noisy environment. It also deploys collision detection sensors, providing 360° protection from safety incidents on the factory floor. Its characteristic accuracy and flexibility make it a perfect replacement for bulky, stationary conveyors or gantries on assembly lines.

To accomplish the task assigned to it, the AGV steers automatically. AGVs are classified into bidirectional, unidirectional, multi-load, and unit-load types. Whereas bidirectional AGVs can drive in both directions along the same guided path, unidirectional AGVs can only advance along a guided approach in three ways [3], [4], or [5]. While a single load is transported from one work centre to another using unit AGVs, a multi-load AGV system could pick up and deliver many shipments in much less time to any work centre, greatly enhancing the efficiency of the material handling system and the FMS facility [6], [7]. Multi-load AGVs have the potential to improve overall FMS productivity and adaptability. The dispatching and scheduling that include collision-free routing of AGVs for the work centre are done separately or instantaneously. Work centres and AGVs are scheduled concurrently, which can increase MS throughput, achieve optimal material handling resource utilization, and reduce AGV arrival times in either MS.

While simultaneously scheduling task centres and AGVs is primarily complex, the results are more effective than separately scheduling work centres and AGVs. If multi-load AGVs are used for material handling pocess for FMS instead of unit-load AGVs, the scheduling process becomes more complex, and the system's efficiency decreases. A possible research gap is studied from various research works to schedule multi-load AGVs and jobs performed at FMS work centres with the least amount of time travel and waiting time. The primary use of a multi-load AGV in a spinning mill is to drop off the processed byproduct automatically, transport it to the following production department, and then transport the completed yarn to a warehouse for packaging, which can cut down on labour and operational costs [8]. The path prediction strategy of the multi-load AGV in a spinning mill immediately needs to be studied due to the distinct layout and production system.

Presently, [9] has presented an enhanced harmony search algorithm to handle the task assignment and sequencing of multiple AGVs in the multi-load AGV's path planning problem. For multi-robot systems, [10] suggested a resource-based task allocation mechanism. The Dijkstra and Dynamic Time-Window algorithms schedule the shortest and most conflict-free route in the [11] dynamic routing method for the AGV system's fastest path planning and conflict prevention. A modified Particle Swarm Optimization (PSO) method was presented by [12] to search shortest path for a solar-powered Unmanned Ground Vehicle (UGV). A routing method using time windows having a vectorial shape was proposed by [13]. The technique is vulnerable to supplementary scheduling and routing issues because of the use of time windows.

In order to create the path planning model, [14] used the scheduling policy driven by actions like the research method, a mixed-integer programming model for the scheduling of the multi-load AGV, the minimum total operational costs as the planning target, operational constraints, the size of the time window, as well as load balance as constraint conditions. Reference [15] developed a finite element model with such a new objective function that combines two indicators, namely the standard deviation of the overall travel distance of an AGV and the waiting time of computer static control material buffers. Reference [16] examined a multi-objective optimization method that measured time costs, the bare minimum of vehicles, and work schedules. Reference [17] suggested a technique for deciding on the optimal path in a flexible job shop with MS, considering two time and cost factors. Reference [18] has used a mathematical programming method to choose the optimal route for AGV transportation based on the BOM while assuming time, costs, and AGV capabilities as a triple standard.

Much research about the path planning of AGVs, including static and dynamic planning, has been conducted over the past few decades by academics at home and abroad. A multi-AGV static path model was created by [19] with a time window to minimise the maximum task volume crossed by each edge inside the simulation graph. A direct and practically continuous trajectory generation system was put out by [20] to produce the shortest collision-free path for multi-AGV systems automatically. The techniques mentioned above exclusively used static path planning, had weak dynamic adaptability and failed to take operation time and path length into account, substantially reducing conflicts. The author [21] applied dynamic path planning to overcome the drawbacks of static path planning and developed a conflict-free path approach for shortest path planning that relied on the evolutionary algorithm, although this technique had an excessively long total travel time. The running path and state scheduling of each AGV were likewise planned by [22] using the frog leaping model. Moreover, this method concentrated on preventing collisions and failed to consider real-time applications, avoiding obstacles or flaws like low system efficiency while collisions occurred continuously.

Several factors, such as guide-path design, AGV scheduling, AGV fleet size, idle-vehicle positioning, vehicle routing, and battery maintenance, are considered while implementing an AGV system [23]. AGV scheduling is broken down into two issues: dispatching, routing, and scheduling. Dispatching includes task selection and assignments for AGVs; sequencing such tasks involves choosing specific paths for reaching assigned destinations; scheduling involves figuring out the arrival times and the charging period of AGVs [9]. The shipping and scheduling of multi-load AGVs are the

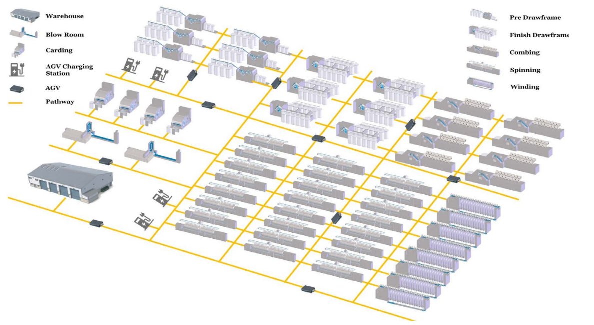


FIGURE 3. AGV path layout in a spinning mill.

main topics of this research (commonly termed ''scheduling multi-load AGVs'').

The dynamic nature of MS was considered by [24], who proposed the multi-agent-based system for instantaneous scheduling in flexible machine groups and material handling. The agents in the model are self-sufficient and can work together and talk with other agents in the system.

Because of this, the scheduling problem has become harder to handle and more complicated as it has grown. Due to its effective global optimization capability and significant flexibility, the intelligent optimization algorithm is applied to solving the AGV scheduling problem. The ant colony algorithm and optimization are used to solve the AGV multi-objective optimization scheduling problem to tally workloads across vehicles, material transport consumption time, and AGV use rate. In order to resolve the multi-objective AGV scheduling optimization problem, Maryam [16] suggested a hybrid between genetic algorithms and PSO.

To solve the scheduling problems in the AGV environment, this paper proposes the Local Search Probability based Memetic Water Cycle (LSPM-WC) algorithm—a combination of the Water Cycle Algorithm (WCA) and the Local Search Probability based Memetic Algorithm (LSPMA)— to find the best scheduling options for multiple AGVs employed in the spinning mill industry.

The proposed paper is organized as follows: Section II presents the problem description; Section III discusses the proposed methodology; the experimental results and analysis are presented in Section IV; the work is concluded in section V.

II. PROBLEM DESCRIPTION

Figure 3 depicts the spinning mill industry's layout and the multi-load transportation path. The multi-load Automated Guided Vehicle (AGV) departs from the warehouse but travels along the path node corresponding to each processing department using a feasible approach. The AGV automatically loads the bales and then transfers unopened bales towards the blow room. The AGV handles all loads from one department to the next and inside the floor after unloading the bales and returning to the blow room workshop to transport the finished product towards the following production department.

A. ASSUMPTIONS AND LIMITATIONS

The following assumptions are considered:

Depending on the different setups used, the type, as well as the operation of a selected FMS environment, varies. As a result, the system setup is accurately determined before AGV scheduling. The following sections outline the system setup, underlying assumptions, and objective criteria used in this study.

(i) A path between two points is not always identical; a system controller could alter an AGV's route to one at the predefined point if the next lane or junction is congested.
(ii) The work centres and locations remain unchanged until all the jobs under consideration are completed.
(iii) AGVs would not experience delays inside a buffer area while awaiting jobs; the AGVs, and the machines, will constantly run without interruption.
(iv) The fleet consists of G=1, 2,...|G| AGVs. Each AGV transports a specific task. The AGVs are usually considered empty at the start of the process.
(v) All AGVs are reliable, carry a certain weight per unit, and move at a steady average speed.
(vi) Both the L/U point-to-machine and machine-to-machine distances are determined.
(vii) No traffic problems, deadlocks, collisions, or conflicts exist, and no jobs are being preempted.
(viii) Pallets, input and output buffers, and loading and unloading (L/U) equipment are allotted appropriately.
(ix) At any assumed time, only one product is used per machine.
(x) The processing time for a job includes the time it consider to load and unload it onto the machine.
(xi) All procedures assume that the setup time on the work centre is '0'.
(xii) AGVs could always park in the designated parking area or drop-off point (DOP).
(xiii) The AGVs are positioned in the transit or charging port to load or unload materials from one source to the destination.

B. PROBLEM STATEMENT

The objective of this scheduling problem would be to implement the AGVs in a way that satisfies all specified criteria and results in an optimum schedule with the minimum travel and waiting time possible. The AGV in this scenario could be located anywhere within the spinning mill industry. At the charging port or while transiting between its source and destination, it can be accessible.

(i) Let t denote the volume of work
(ii) Node n and node $t+n=$ the select and delivery place of the n^{th} task inside the network
(iii) Inside the network, multiple nodes are denote the similar physical location of the production department
(iv) A node-set is formed by computing node 0 and (2t+1) to be an AGV's actual starting and endpoint towards a network.

$$T = \{0, \ldots, t, t+1, t+2 \ldots, 2t, 2t+1\}.$$

(v) The choice, and delivery points, are, correspondingly, comprised of two sets as

$$L^{+} = \{1, 2 \ldots, t\}$$
$$L^{-} = \{t+1, t+2 \ldots, 2t.\}$$
$$L = L^{+} \cup L^{-}$$

(vi) L = Set of nodes from which to select and delivery location other than an FMS facility.
(vii) The variables for the notations used in the following equations have been worked out.

j, Index for Tasks; $j = 1, 2, \ldots, x$
g, Index for AGVs; $g = 1, 2, \ldots, t$
d, d' Index for Processing Centre (PC); $d = 1, 2, \ldots, y$; $d' = 2, 3, \ldots, y$
M_j = Selection time interval of the j^{th} job
M_{gy} = Time at which the AGV, g, sub-nodes the PC
C_g = Size of AGV, g
r = Number of jobs at a task center
l_{jd} = Processing time of job "j" being processed at PC "d"
p_j = The task's limit "j" being processed at PC "d"
F_{jd} = Deadline time for job "j" being processed at PC "d"
D_{nm} = Travel time from the physical position of node n, $\mathbb{P}_n$ to the physical position of node m, $\mathbb{P}_m$.
Z_{nmg} = Measure of AGV, "g" from node "n" to "m".
If, $Z_{nmg} = 1$; AGV "g" moves from node "n" to "m"; else $Z_{nmg} = 1$. So, its domain is $\{0, 1\}$.
$\mathbb{L}$ = Load
$\mathbb{L}_{gn}$ = Load at AGV "g" when it sub-nodes "n".
Initially $\mathbb{L}_{gy} = 0$
M_{gn} = Time at which the AGVg starts service at node "n"
$\mathbb{M}$ = Makespan
υ = Usage
Initially $M_{gy} = 0$
$\alpha; \beta; \delta;$ = time-related weights applied to the objective function

From EQU (1) to EQU (14), constraints and objective functions of an issue have been recognized. The load carried by the AGV when it departs from the first pickup point (PP) following the work centre and travels to any other PP or DOP is depicted by equations. An AGV's load will increase or decrease by one if it travels to a PP or DOP, so after crossing the first PP or DOP, the AGV in a material handling system functions with multiple loads. The scheduling problem to multi-load AGV presented within this work is an NP-hard problem based upon satisfying constraints and optimised while considering the minimum travel and waiting time model.

AGV first leaves the work centre before following a PP. AGVs can travel towards whichever pickup or delivery point they choose. Before delivery to the last work centre, a multi-load AGV delivers a job towards the second-last work centre. The service time of the current node and the travel time between existing and present nodes are considered when calculating the starting service time at each node.

AGV pickup and delivery points are subject to restrictions. An AGV must leave a node after entering it, and if it visits the pickup node, it also visits a related delivery node. This requirement confirms that every PP is to be visited through an AGV. EQU (1) to EQU (14)

$$\left(Z_{0gl}=1\right) \Rightarrow \mathbb{L}_{gm}=1;\, g \in G_m \in L^{+} \tag{1}$$

$$\left(Z_{nmg}=1\right) \Rightarrow \left\{\begin{matrix} \mathbb{L}_{gm}=\mathbb{L}_{gn}+1;\, g \in G, m \in \mathrm{L}^{+}, n \in L, n \neq m \\ \mathbb{L}_{gm}=\mathbb{L}_{gn}-1;\, g \in \mathrm{G}, m \in \mathrm{L}^{-}, n \in L, n \neq m \end{matrix}\right\} \tag{2}$$

$$\left(Z_{0gl}=1\right) \Rightarrow \mathrm{M}_{gm}=\mathrm{M}_{g0}+\mathrm{M}_{\mathbb{P}_0,\mathbb{P}_m},\, m \in \mathrm{L}^{+}, g \in G \tag{3}$$

$$\left(Z_{nmg}=1\right) \Rightarrow \mathrm{M}_{gm} = M_{gn}+\mathrm{M}_{\mathbb{P}_n,\mathbb{P}_m},\, \mathrm{n}, \mathrm{m} \in \mathrm{L}, g \in G \tag{4}$$

$$\left(Z_{n(2t+1)g}=1\right) \Rightarrow \mathrm{M}_{g(2t+1)} = M_{gn}+\mathrm{M}_{\mathbb{P}_n,\mathbb{P}_{(2t+1)}},\, n \in \mathrm{L}^{-}, g \in G \tag{5}$$

$$\sum\nolimits_{g\in G}\sum\nolimits_{m\in T} Z_{nmg}=1,\, n \in \mathrm{L}^{+} \tag{6}$$

$$\sum\nolimits_{m\in T} Z_{nmg}-\sum\nolimits_{\mathrm{jeN}} Z_{mng}=1,\, n \in L, g \in G \tag{7}$$

$$\sum\nolimits_{m\in T} Z_{nmg}-\sum\nolimits_{\mathrm{jeN}} Z_{m(t+1)g}=1,\, n \in \mathrm{L}^{+}, g \in G \tag{8}$$

$$\sum\nolimits_{m\in \mathrm{L}^{+}} Z_{0gl}=1,\, g \in G \tag{9}$$

$$\sum\nolimits_{m\in \mathrm{L}^{-}} Z_{n(2t+1)}=1,\, g \in G \tag{10}$$

$$\mathbb{L}_{gn} \leq \mathrm{C}_g,\, \mathrm{g} \in \mathrm{G}, n \in L$$

$$min\,\text{time} = \sum\nolimits_{\mathrm{g}\in\mathrm{G}}\left\{\alpha\sum\nolimits_{n\in L}\sum\nolimits_{m\in L_{m\neq1}} Z_{nmg}\cdot \mathrm{M}_{nm} + \sum\nolimits_{r}\left(\beta\sum\nolimits_{n\in L}\left|\mathrm{M}_n-\mathrm{M}_{\mathrm{gn}}\right|^{+} + \delta\sum\nolimits_{n\in L}\left|\mathrm{M}_{\mathrm{gn}}-\mathrm{M}_n\right|^{+}\right)\right\} \tag{11}$$

$$\alpha=\sum\nolimits_{d=1}\sum\nolimits_{j=1}\left|l_{jd}-p_{jd}\right|^2 \tag{12}$$

$$\beta=\sum\nolimits_{d=1}\sum\nolimits_{j=1}\left|l_{jd}-F_{jd}\right|^2 \tag{13}$$

$$\delta=\sum\nolimits_{d=1}\sum\nolimits_{j=1}\left|l_{jd}-C_{jd}\right|^2 \tag{14}$$

1) MINIMIZING THE MAKESPAN

This phase measures the makespan ($\mathbb{M}$), which is the duration needed for all tasks to be executed. Makespan is expressed by EQU (15):

$$\mathbb{M}=\max\{\left(Z_{nmg}+l_{jd}\right)\} \tag{15}$$

2) FITNESS VALUE (FV)

Assume FV is used to estimate the authority of task scheduling. Load balancing knowledge is expected to be required for effective AGV deployment. The fitness function for assigning an actual task to an AGV is calculated using EQU (16).

$$FV\,(g)=\frac{1}{\mathbb{M}}\times\max(\upsilon) \tag{16}$$

Considering the machine's varying production times, the average usage of an AGV is premeditated. The operation of the distinct device is evaluated by using EQU (17)

$$\upsilon\left(g_{jd}\right)=\frac{F_{jd}}{\mathbb{M}} \tag{17}$$

3) OBJECTIVE FUNCTION (f)

The objective function of this phenomenon is the average total execution time of all tasks that are given to the AGVs, and it is found using the EQU (18). G is taken into account as multiple AGVs in this situation. This value represents the equilibrium between AGV use and the optimal schedule.

$$f=min\left\{\frac{\sum_{i=1}^{g} FV\,(g)}{g}\right\} \tag{18}$$

Every AGV's first visit is to a pickup node, while its last is to a delivery node. When AGV "g" departs from node "n," its load cannot be more than its carrying capacity. The total travel time of the AGVs is computed while considering their waiting time and their delay time to perform their tasks. AGV's waiting time and delay time will affect the objective function if they are positive.

III. PROPOSED METHODOLOGY

A. LOCAL SEARCH PROBABILITY-BASED MEMETIC ALGORITHM (LSPMA)

By considering some significant MA-related problems, possible algorithmic MA improvements are accomplished. Assuming a local search frequency, or how frequently the local search occurs, is just one of the initial issues relevant to a MA model.

Using the problem-specific local search schemes, select a suitable solution to such a random, definite local search direction. Also, use only local search on your chosen solution to make local search more effective. While reducing the number of solutions over that local search, this article uses the local search probability or LSP. According to the present local search direction, the proposed algorithm decides.

The suggested method reduces the production time of a task scheduling problem, where the pseudocode is shown in Algorithm 1, by using the local search probability, or LSP, for choosing individuals during the next population.

1) GENETIC REPRESENTATION

Each task graph node's start time and AGV allocation make up the schedule. So, every chromosome is denoted as a group of genes, or a task-AGV pair (t_i, g_i), which denotes that

Algorithm 1 LSPM Algorithm

Step 1:	Use population size "*PS*" as a metric to create chromosomes and measure their "*FV*" ;
Step 2:	Set initial solution, $Inis := max(FV)$;
Step 3:	Perform LSP over chosen solutions to determine the best fit value. "LS_{best}";
Step 4:	Update$Inis := LS_{best}$;
Step 5:	While (True)
Step 6:	Crossover ()
Step 7:	Mutation ()
Step 8:	Execute STEP 3 over again from the population acquired from STEP 7
Step 9:	If $FV < LS_{best}$
Step 10:	Update LS_{best}
Step 11:	End If
Step 12:	End While

FIGURE 4. Chromosomal representation.

task t_i is associated with the processor g_i in Figure 4. The positioning of genes on chromosomes regulates the sequence in which tasks must be carried out.

2) INITIALIZATION

Thus, the generation of the initial population is based upon the precedence calculation of tasks at all levels. A chromosome's fitness is inversely proportional to the length of an associated schedule because the goal of a task-scheduling problem is to locate the shortest program possible. Here, the reduced task completion time is used to find FV.

3) SELECTION

During this step, a population's chromosomes are ranked from best to worst in terms of FV. They are then selected for the pool.

4) RECOMBINATION

New chromosomes are produced by joining two selected parent chromosomes. The second component of each chromosome is swapped at a randomly chosen point. A crossover probability is set randomly when alternating between one-point and two-point crossings.

5) CROSSOVER

By swapping some genes between two chosen chromosomes, the crossover operator creates two new chromosomes for the following generation. This investigation uses two crossover operators—one-point crossover and two-point crossover—based on the interchange of partial strings. Based on the Crossover Rate (CR) and Population Size (PS), the number of crossovers is computed as follows in EQU (19):

$$No.of\ crossovers = \frac{(CR \times PS)}{2} \tag{19}$$

6) MUTATION OPERATOR

Compared to the crossover operator, such an operator is active with a lower probability (0.1 or less). Its objective is to prevent the state search from settling on a locally optimal solution. The population gradually gets fitness by changing a randomly chosen gene (t_i, g_i) to (t_i, g_j) on each chromosome in the fittest individuals. As a result, the population steadily gets better. The proposed algorithm does not set a predetermined probability of crossover and partial-gene mutation.

7) LOCAL SEARCH

During the local search phase, a neighbour is created at random from the neighbourhood of a current solution. A recent solution is substituted if the neighbour is superior, and a local search for a new current solution is carried out similarly. When an offspring's quality is deficient, applying local search looks like an unwanted computational burden. As a result, only good offspring are used for local searches.

8) TERMINATION CRITERIA

The algorithm ends if no enhancement solution is observed during the last '*n*' iterations. This ranges from 50 to 500 depending on the level of the problem and the required optimization of a solution.

B. WATER CYCLE ALGORITHM (WCA)

The hydrologic cycle technique, which has five phases and is based on the nature-inspired algorithm WCA, was first presented in [25]. Such as (a) transpiration (water from lakes and rivers evaporates at the same time that plants release water through photosynthesis), (b) condensation (cloud formation in the air, which then condenses and cools), (c) precipitation (discharge of water to earth similar to ice melting), (d) percolation (groundwater, also known as reserved water in a field), and (e) evaporation (evaporation converts underground water and releases it to form a stream). The proposed method begins with a total population known as raindrops produced by rain or other precipitation.

This paper initially anticipates downpours or other precipitation. This research work shows that rivers are formed when raindrops are connected to form streams. A portion of the streams also goes directly into the ocean. Each river and stream flows into the sea, an ideal spot. Figure 5 shows the flow of the streams towards a specific river, with the star representing the river and the circle representing the stream, respectively.

At first, raindrops are generated at random, and the integrity of the schedule is selected by computing good

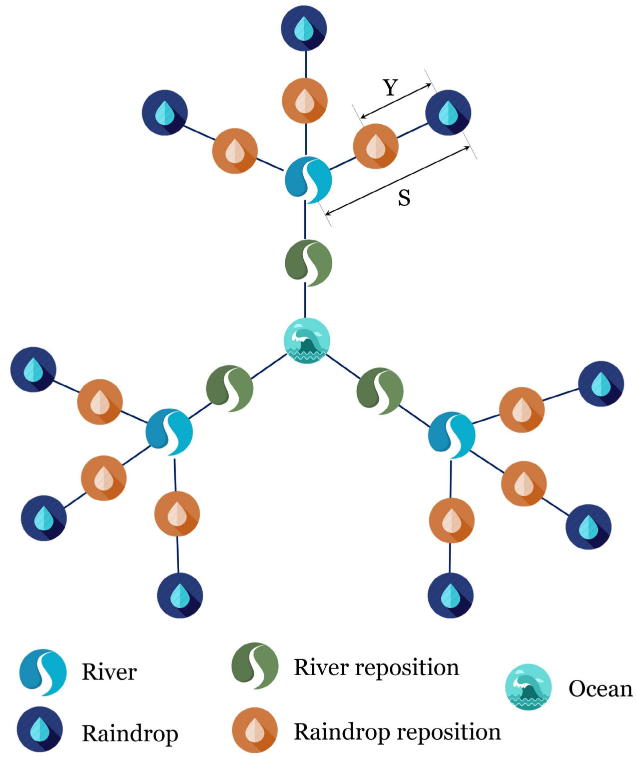


FIGURE 5. The water cycle: from raindrop to stream, then the river, and finally ocean.

raindrops. First of all, we acknowledge that there have been downpours or precipitations. The best person is chosen to represent an ocean as a raindrop. At that moment, a few heavy raindrops are selected to represent the river, and the remaining raindrops are considered streams that flow into rivers and oceans. After repeatedly running a relatively similar system for the predetermined cycles, an optimal solution is found. The *"guidance points"* in the suggested technique, as shown in Figure 5, are the rivers and other well-select places aside from the best one (ocean), which are used to steer other member populations in the right direction. Rivers change direction and flow into the sea. The empty white outlines depict the shifting of rivers and streams (raindrops). To simplify the analysis, the WCA steps are presented in algorithm 2.

Three job subsequence options for every multi-load AGV are used to provide a neighbourhood solution for such a WCA.

(i) With each multi-load AGV, the sub-sequence of job relocation: Incorporates a specific set of tasks performed by an AGV from one route to another.
(ii) Exchange or a subsequence of tasks between two different paths improves the response to the AGV routing problem for each multi-load AGV.
(iii) This work combines the two methods above, the subsequence of the job mixture to every multi-load AGV attempt for relocating, exchanging or subsequence of

Algorithm 2 For WCA

Step 1: Choose "raindrops" as the original variables. The raindrops' function inside WCA is analogous to a "chromosome" inside GA or a "particle position" inside PSO. A raindrop is stated by one of the following methods:

$$A\ streamorRaind\ rop = \{x_1, x_2, x_3, \ldots x_n\} \tag{20}$$

Step 2: Consider N_{pop} as the number of raindrops and N_{var} as the number of variables, using EQU (20) creates a random initial population.

$$R_{pop} = \begin{bmatrix} raindrops_1 \\ raindrops_2 \\ raindrops_3 \\ \vdots \\ raindrops_{N_{pop}} \end{bmatrix} = \begin{bmatrix} x_1^1 & x_2^1 & \ldots & x_{N_{Var}}^1 \\ x_1^2 & x_2^2 & \ldots & x_{N_{Var}}^2 \\ \ldots & \ldots & \ldots & \ldots \\ x_1^{N_{pop}} & x_2^{N_{pop}} & \ldots & x_{N_{Var}}^{N_{pop}} \end{bmatrix} \tag{21}$$

Step 3: Calculate the raindrops' fitness using: EQU (22)

$$Cost_i = f\left(x_1^i, x_2^i, \cdots, x_{N_{Var}}^i\right) i = 1, 2, 3, \cdots, N_{pop} \tag{22}$$

Step 4: The best N_{rs} individuals are chosen to represent rivers and the ocean out of N_{pop} raindrops. Figure out the number of rivers using EQU (23)

$$N_{\mathrm{rs}} = No.of\ Rivers + 1 \tag{23}$$

Step 5: The best raindrop is taken to represent the ocean, and the rest of the population is considered to be streams or raindrops flowing into the river and ocean using EQU (24):

$$N_{Raindrops} = N_{pop} - N_{\mathrm{rs}} \tag{24}$$

Step 6: Considering the number of streams. "NS_n", each one streams into a specific river and ocean. The following EQU (25) is used to compute the flow intensity of rivers and oceans

$$NS_n = round\left\{\left|\frac{Cost_i}{\sum_{i=1}^{N_{rs}} Cost_i}\right| \times N_{Raindrops}\right\}, \quad n = 1, 2, \ldots, N_{rs} \tag{25}$$

An individual stream's separation from a specific river that it flows into is analysed by EQU (26)

$$Y \in (0, L \times s), L > 1 \tag{26}$$

where between 1 and 2 or L = 2 is the user-defined value. s is the current distance in the middle of streams as well as the river. Y is the number between 0 and (LxS) in each distribution.

Algorithm 2 *(continued.)* For WCA

Step 7: EQU (27) determines the rate at which streams flow into rivers. Rivers and streams' new locations are computed as follows:

$$Y^{i+1}_{Stream} = Y^{i}_{Stream} + rand \times L \times \left(Y^{i}_{River} - Y^{i}_{Stream}\right) \tag{27}$$

Step 8: In order to determine the rivers' flow into the ocean (the most extraordinary downhill location): EQU (28)

$$Y^{i+1}_{River} = Y^{i}_{River} + rand \times L \times \left(Y^{i}_{Sea} - Y^{i}_{River}\right) \tag{28}$$

where the uniformly distributed random number is 0 and 1.

Step 9: To find the best global solution G_{best}, exchange a stream's position with the river.

Step 10: While the river fixes a superior solution to the sea, the location of a river is traded, similar to Step 7.

Step 11: Analyse the evaporation conditions using EQU (29)

$$if \left|Y^{i}_{Sea} - Y^{i}_{River}\right| < s_{max}; i = 1, 2, \ldots, N_{rs} - 1 \tag{29}$$

Wherein a smaller value that regulates a search depth near an ocean is $s_{max} \cong 0$ used. The value of s_{max} drops decrease with each iteration because evaporation decreases as it rains, EQU (29)

$$s^{i+1}_{max} = s^{i}_{max} - \frac{s^{i}_{max}}{\max iteration} \tag{30}$$

Step 12: Lower bound values are denoted by the symbol "LB". The assumed problem characterises the upper determined value or UB. In specific locations identified using EQU (31), newly created raindrops produce new streams arbitrarily.

$$Y^{new}_{Stream} = LB + rand \times (UB - LB) \tag{31}$$

The algorithm's optimum point's computational execution and convergence rate are defined by EQU (32)

$$Y^{new}_{Stream} = Y_{\text{sea}} + \sqrt{\mu} \times \text{randn}\left(1, N_{\text{var}}\right) \tag{32}$$

Step 13: To lessen the value of s_{max}, EQU (33)

Step 14: Verify the convergence criteria. The process stops if the halting criteria are met; otherwise, it will return to Step 7.

tasks, and concurrently assessing the options to discover the option providing less travel and waiting time.

The subsequence task mix sequence for each AGV was looked at in this paper because it led to a better solution.

C. TASK SCHEDULING MULTI-LOAD AGV USING LSP-BASED MEMETIC AND WCA

A technique's performance is determined by its capacity for solution exploration (global search) and solution exploitation (local search). The WCA places less emphasis on the exploitation process than on the exploration process because the raindrops under consideration converge prematurely to specific locations during an early phase due to their attraction as globally best positions. Using evolutionary algorithms that have proven helpful in many application areas improves an algorithm's capability to use local search data for feasible outcomes. In order to generate the best solutions, local search possibilities within optimization are improved. This research investigation integrates and applies the WCA to the optimal global solution and the LSPMA to the optimal local solution. It is done in the spinning mill industry to schedule multi-load AGVs with less travel and waiting time. LSPM-WC is the name of the new algorithm 3 proposed. By integrating the solutions in a manner akin to how a crossover is used in GA, the LSPM-WC enhances local search capabilities. A random selection of a population of such solutions is subjected to a recombination process. Recombining the frequent results in improved FVs than the initial or prior values; therefore, the new solutions are implemented for the existing solutions. The flowchart shows that the algorithm produces reasonable solutions with less travel time and waiting time, as shown in Figure 6 because the newly combined solutions produced by adequate local search balance the algorithm's exploitation and exploration processes. For the scheduling of multi-load AGVs, LSPM-WC produces the best possible results with less travel and waiting time.

The following three techniques are taken into account to find the best solution for the WCA and LSPM-WC:

Algorithm 3 LSPM-WC Algorithm

Step 1:	Initialize the raindrop population, R_{pop}
Step 2:	Using EQU (18), measure the objective function for scheduling
Step 3:	Update the local search solution. L_{best} and the global search solution G_{best}
Step 4:	Employ the LSP-based memetic algorithm using the initial population and objective function;
Step 5:	For the initial population, recombine $p\%(R_{pop})$
Step 6:	Calculate new L_{best} and G_{best}
Step 7:	Compare the initial solution with the new search solution
Step 8:	If new L_{best} and G_{best} is better
Step 9:	Replace the search solutions with new values
Step 10:	Else
Step 11:	Retain the initial solutions
Step 12:	Update Scheduling, L_{best} and G_{best}
Step 13:	If the termination condition does not arrive
Step 14:	Go to Step 4
Step 15:	Else
Step 16:	End

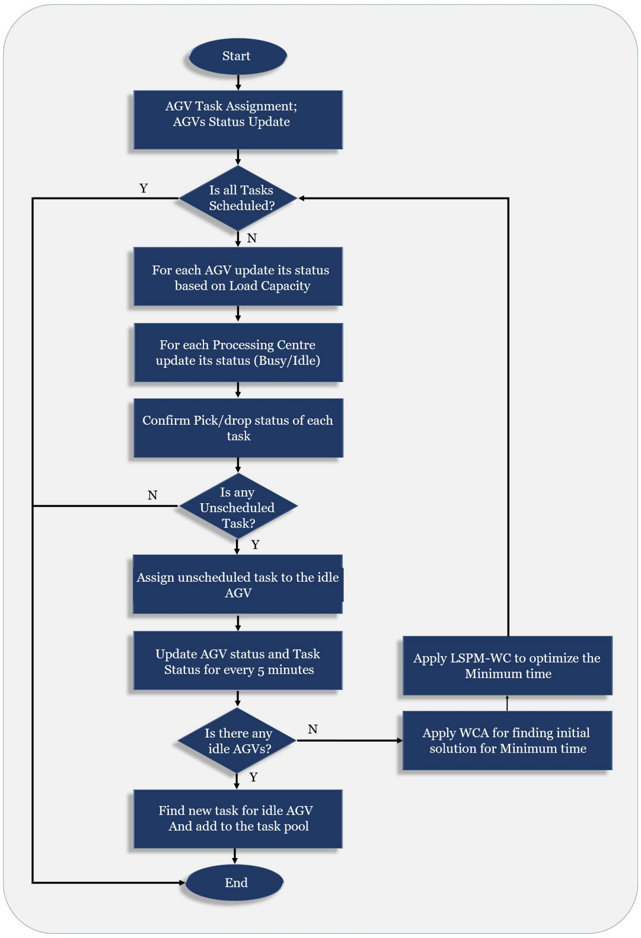


FIGURE 6. LSPM-WC algorithm flowchart.

1) WCA'S INITIAL DETERMINISTIC FEASIBLE SOLUTION

In this scenario, the number of tasks divided by the total volume of AGVs determines the travel time for each AGV.

2) WCA'S INITIAL RANDOM FEASIBLE SOLUTION

To accept the principles of probability, a few random tasks are selected. With this method, the process commences in various neighbourhoods.

3) SOLUTIONS FROM LSPM-WC

Following the generation of the initial optimal solution through WCA for the single-loaded condition of AGVs, LSPM-WC is once again applied to obtain a better solution for multi-load AGVs.

IV. RESULT AND ANALYSIS

A. EXPERIMENTAL SETTING

This section presents the experimental results. The algorithms in this comparison are all implemented in MATLAB version 2021a and run on a personal computer with an Intel Core i7 3610QM CPU running at 2.5 GHz, 16 GB of RAM, and Microsoft Windows 11 installed as the operating system. Comparisons against many algorithms are made to confirm the efficiency of the proposed model.

TABLE 1. Time travelled by AGVs between Pickup and Drop locations (Example 1).

	Pickup	M_1	M_2	M_3	M_4	M_5	M_6	M_7	M_8
Drop	0	1	5	8	10	13	15	17	20
M_1	21	0	4	7	9	12	14	16	19
M_2	19	20	0	3	5	8	10	12	15
M_3	17	18	22	0	2	5	7	9	12
M_4	14	15	19	22	0	3	5	7	10
M_5	10	11	15	18	20	0	2	4	7
M_6	8	9	13	16	18	21	0	2	5
M_7	5	6	10	13	15	18	20	0	3
M_8	2	3	7	10	12	15	17	19	0

TABLE 2. Time travelled by AGVs between Pickup and Drop locations (Example 2).

	Pickup	M_1	M_2	M_3	M_4	M_5	M_6	M_7	M_8	M_9	M_{10}	M_{11}	M_{12}
Drop	0	11	14	17	9	10	7	12	13	21	25	18	22
M_1	11	0	9	15	5	14	20	16	23	6	19	11	17
M_2	14	9	0	6	17	25	9	6	11	15	21	12	19
M_3	17	15	6	0	18	8	12	20	12	5	23	15	9
M_4	9	5	17	18	0	14	4	23	10	9	18	13	21
M_5	10	14	25	8	14	0	6	10	15	12	21	4	18
M_6	7	20	9	12	4	6	0	17	19	11	7	10	13
M_7	12	16	6	20	23	10	17	0	9	18	25	14	10
M_8	13	23	11	12	10	15	19	9	0	11	18	5	23
M_9	21	6	15	5	9	12	11	18	11	0	14	11	18
M_{10}	25	19	21	23	18	21	7	25	18	14	0	17	20
M_{11}	18	11	12	15	13	4	10	14	5	11	17	0	25
M_{12}	22	17	19	9	21	18	13	10	23	18	20	25	0

TABLE 3. Processing time for each process on each machine (Example 1).

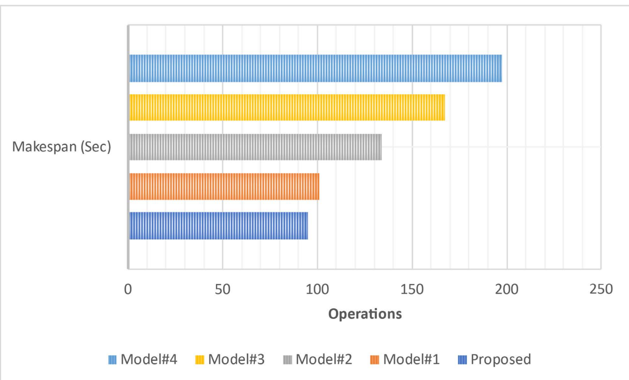
Task	Operation	Machine	Duration	Task	Operation	Machine	Duration
1	1	M_1	11	5	1	M_7	21
	2	M_2	14		2	M_8	17
	3	M_4	15		3	M_4	14
2	1	M_2	29		4	M_2	19
	2	M_3	24	6	1	M_5	16
	3	M_5	10		2	M_7	9
	4	M_6	12		3	M_8	21
3	1	M_5	20	7	1	M_7	24
	2	M_6	10		2	M_2	20
	3	M_8	16		3	M_3	10
4	1	M_5	17		4	M_5	12
	2	M_6	5	8	1	M_8	19
	3	M_1	24		2	M_2	21
	4	M_2	21		3	M_3	30
	5	M_7	4		4	M_4	14

Two static tests were used to validate the model. The first example involves eight machines ($M_1, \ldots, M_8$) Processing 8 tasks ($t_1, \ldots, t_8$), each of which required multiple operations. The second example has 12 machines ($M_1, \ldots, M_{12}$) processing 16 tasks ($t_1, \ldots, t_{16}$), with each task requiring multiple operations. The AGV's travel time between pickup and dropping points and machines is shown in Table 1 and Table 2, and the processing time for each machine operation is shown in Tables 3 and 4 for both examples.

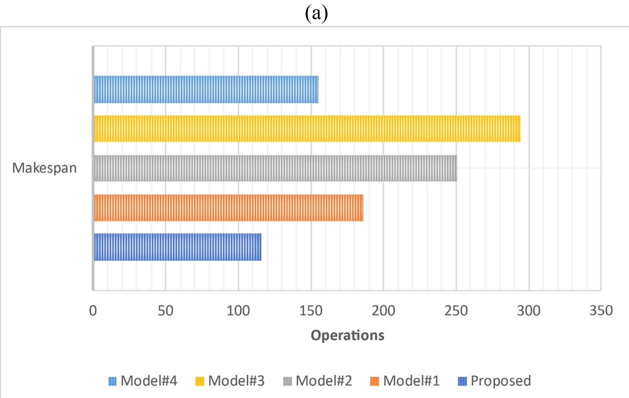
According to the experimental study, the crossover and mutation rates were 0.2 and 0.08 for all tests, and the starting variables for the WCA were selected as 50, 8 and 1×10^{-5},

TABLE 4. Processing time for each process on each machine (Example 2).

Task	Operation	Machine	Duration	Task	Operation	Machine	Duration
1	1	M_1	24	7	1	M_{11}	5
	2	M_4	10		2	M_1	24
	3	M_6	12		3	M_3	21
	4	M_8	20	8	1	M_9	12
2	1	M_2	10		2	M_{11}	19
	2	M_5	16		3	M_{12}	21
	3	M_{10}	9		4	M_3	30
3	1	M_3	21	9	1	M_4	10
	2	M_7	24		2	M_7	12
	3	M_{11}	20		3	M_8	20
4	1	M_1	10		4	M_{10}	10
	2	M_3	14	10	1	M_{12}	16
	3	M_7	15		2	M_1	17
	4	M_{12}	49		3	M_3	9
5	1	M_3	24		4	M_6	11
	2	M_5	10	11	1	M_7	14
6	1	M_2	12		2	M_8	11
	2	M_5	20		3	M_{10}	24
	3	M_9	9		4	M_{11}	5
	4	M_{11}	11	12	1	M_3	11
	5	M_1	14		2	M_5	19

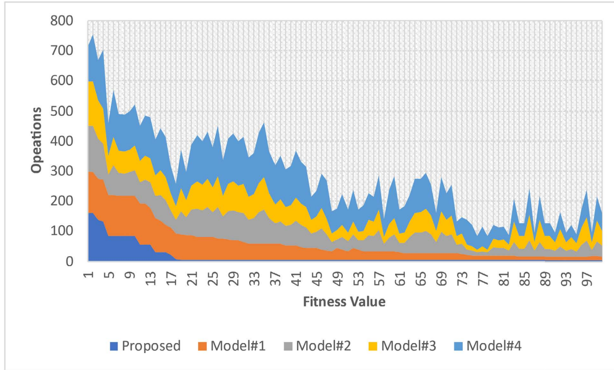


(a)

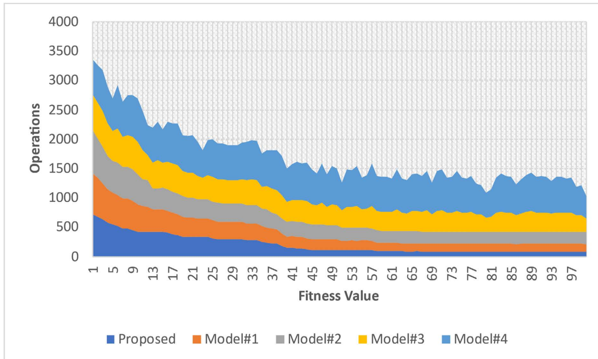


(b)

FIGURE 7. (a): Makespan comparison (Example 1) (b): Makespan comparison (Example 2).



(a)



(b)

FIGURE 8. (a): Fitness comparison (Example 1) (b): Fitness comparison (Example 2).

for N_{pop}, N_{rs}, andS_{max}. The programs are run 50 times by a population size of 100 within 100 iterations. The proposed model is evaluated compared to other task scheduling algorithms based on GA, Swarm Intelligence, and Hybrid.

MODEL#1: "An improved PSO-based task scheduling AGV in a resource-constrained environment," proposed by [15].

MODEL#2: "A scheduling algorithm based on hybrid GA and PSO for employing AGV in having workshops" proposed by [26].

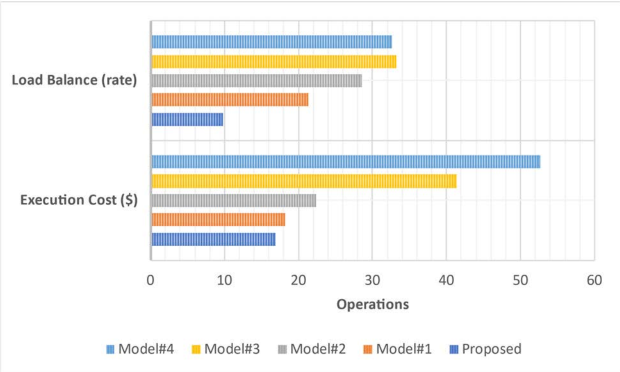
MODEL#3: "A harmony search based AGV scheduling for material transfer in a real-world MS" proposed by [27].

MODEL#4: "A GA + ACO based AGV task scheduling in FMS" proposed by [28].

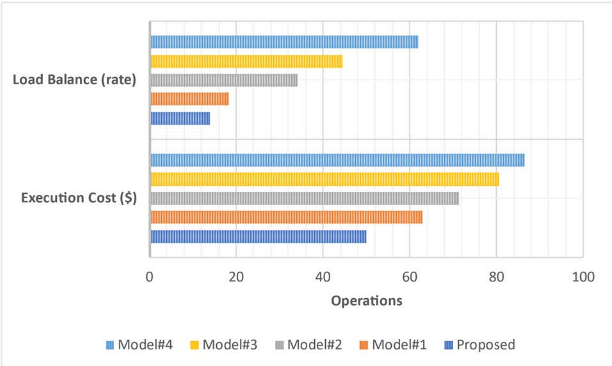
The makespan results for both examples of all the models are presented in Figures 7 (a) and 7 (b). It is visible that Model#1, Model#2 and Model#3 have reduced makespan efficiently. But when compared with the proposed model, the proposed LSPM-WC algorithm outperforms all the other models. EQU 16 has been used to determine the FV and the results from Figures 8 (a) and 8 (b); it is inferred from both examples that the proposed model has better performance than the other models.

ETC and load balance values are shown in Figures 9 (a) and 9 (b), demonstrating the performance results of all the compared models and the correlation between ETC and load balance. All processes over the fitness function show that the proposed model improved overall efficiency on all points of interest more than other modelling techniques.

The optimal task and AGV combination can learn the optimum package transport behaviour. Thus, we're interested in evaluating its behaviour to see if it's sufficient. In addition, this paper optimizes assessing the system's scalability by experimenting with multiple AGVs and observing their

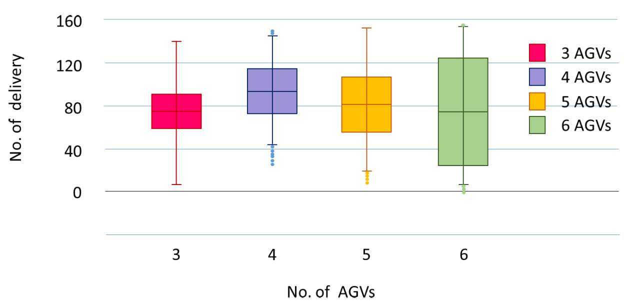


(a)

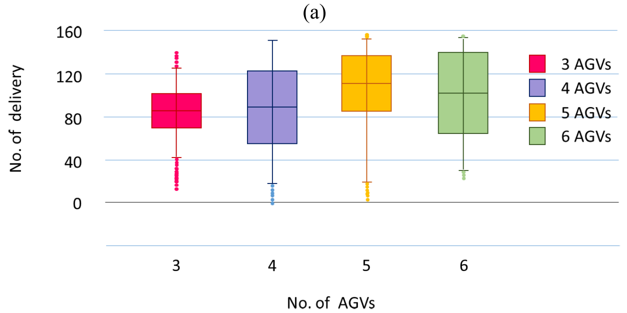


(b)

FIGURE 9. **(a): Execution time cost (ETC) and load balance (Example 1) (b): ETC and load balance (Example 2).**



(a)



(b)

FIGURE 10. **(a) AND 10 (b): Number of deliveries for AGVs in the simulated scenarios A and B 100 episodes of execution has been performed for each configuration.**

behaviour and performance. Given the difficulty of the task at hand, employing a policy at random, like the baseline, which cannot transport any load, is not an option (causing the end of the section within a few simulation steps).

This research work has performed 100 tests for each configuration, such as 3, 4, and 6 cars in the environment, to confirm the effectiveness of these regulations. To balance the traffic flow and increase load delivery speed, the AGVs will be distributed randomly in only one lane. For the used loading and unloading speeds, it is simple to determine that the scheduling model can complete a collision-free product delivery that is close to the theoretical total number of shipments. If we ignore the stop conditions between vehicles, the maximum delivery formula can be illustrated as follows:

$$max_{delivery} = \text{no. of AGVs} \times \frac{no.\ of\ simulation\ steps}{|T_a|} \quad (33)$$

where $|T_a|$ specifies how many cells make up a discretized track. Figures 10 (a) and (b) display the theoretical maximum and average loads sent over 100 executions for different vehicle configurations in the two testing conditions. The number of transported loads increases linearly, as can be seen, and the number of packages delivered is nearly equal to the theoretical maximum. It decodes that the proposed model performs well.

V. CONCLUSION AND FUTURE WORK

In this work, the Local Search Probability Based Memetic Water Cycle (LSPM-WC) algorithm—a combination of the Water Cycle Algorithm and the Local Search Probability Based Memetic Algorithm—has been used to find the best scheduling options for multiple AGVs used in the spinning industry. The results of this proposed work have been evaluated against those of other AGV scheduling algorithms. Despite some work focused on scheduling unit load AGVs and relatively few studies on multi-load AGV task scheduling, this paper study obtains solutions for multi-load AGV task scheduling that result in a minimum makespan for handling materials in a spinning mill; this was achieved using the proposed integrated LSPM-WC algorithm.

When used for material handling in a spinning mill, the LSPM-WC for multi-load AGVs achieves significant throughput. The analysis shows that the LSPM-WC can produce promising results for real-time task scheduling of multi-load AGVs. When designing a path for several AGVs in FMS, it is essential to consider not just the fastest route for the AGVs to take when transporting containers but also whether the crossing or overlapping of AGV driving trajectories causes a collision, congestion, or other conflict issues. If the AGVs' path conflicts aren't resolved, it'll slow down the AGVs' journey time and raise the waiting time in the processing department, both of which will reduce operational efficiency and drive up costs. As a result, one of the primary research guidelines for the future is AGVs' dynamic scheduling mixed with AGVs' active path planning.

REFERENCES

[1] M. Javaid, A. Haleem, R. P. Singh, and R. Suman, "Enabling flexible manufacturing system (FMS) through the applications of industry 4.0 technologies," *Internet Things Cyber-Phys. Syst.*, vol. 2, pp. 49–62, May 2022.

[2] A. C. Godoy and I. G. Pérez, "Integration of sensor and actuator networks and the SCADA system to promote the migration of the legacy flexible manufacturing system towards the industry 4.0 concept," *J. Sensor Actuator Netw.*, vol. 7, no. 2, p. 23, May 2018, doi: 10.3390/jsan7020023.

[3] P. J. Egbelu and J. M. A. Tanchoco, "Potentials for Bi-directional guide-path for automated guided vehicle based systems," *Int. J. Prod. Res.*, vol. 24, no. 5, pp. 1075–1097, Sep. 1986.

[4] R. J. Gaskins and J. A. Tanchoco, "Flow path design for automated guided vehicle systems," *Int. J. Prod. Res.*, vol. 25, no. 5, pp. 667–676, 1987.

[5] R. J. Gaskins, J. M. A. Tanchoco, and F. Taghaboni, "Virtual flow paths for free-ranging automated guided vehicle systems," *Int. J. Prod. Res.*, vol. 27, no. 1, pp. 91–100, 1989.

[6] G. Ulusoy, F. Sivrikaya-Şerifoğlu, and Ü. Bilge, "A genetic algorithm approach to the simultaneous scheduling of machines and automated guided vehicles," *Comput. Oper. Res.*, vol. 24, no. 4, pp. 335–351, 1997.

[7] R. V. D. Meer, *Operational Control of Internal Transport*. IOS Press, Jan. 2000, pp. 1–184.

[8] Q.-V. Dang, N. Singh, I. Adan, T. Martagan, and D. Van De Sande, "Scheduling heterogeneous multi-load AGVs with battery constraints," *Comput. Oper. Res.*, vol. 136, Dec. 2021, Art. no. 105517.

[9] G. Li, X. Li, L. Gao, and B. Zeng, "Tasks assigning and sequencing of multiple AGVs based on an improved harmony search algorithm," *J. Ambient Intell. Humanized Comput.*, vol. 10, no. 11, pp. 4533–4546, Nov. 2019.

[10] D.-H. Lee, "Resource-based task allocation for multi-robot systems," *Robot. Auto. Syst.*, vol. 103, pp. 151–161, May 2018.

[11] T. J. Chen, Y. Sun, W. Dai, W. Tao, and S. Liu, "On the shortest and conflict-free path planning of multi-AGV system based on Dijkstra algorithm and the dynamic time-window method," *Adv. Mater. Res.*, vol. 645, pp. 267–271, Jan. 2013.

[12] N. Smolic-Rocak, S. Bogdan, Z. Kovacic, and T. Petrovic, "Time windows based dynamic routing in multi-AGV systems," *IEEE Trans. Autom. Sci. Eng.*, vol. 7, no. 1, pp. 151–155, Jan. 2010.

[13] A. Kaplan, N. Kingry, P. Uhing, and R. Dai, "Time-optimal path planning with power schedules for a solar-powered ground robot," *IEEE Trans. Autom. Sci. Eng.*, vol. 14, no. 2, pp. 1235–1244, Apr. 2017.

[14] K. G. Huo, Y. Q. Zhang, and Z. H. Hu, "Research on scheduling problem of multi-load AGV at automated container terminal," *J. Dalian Univ. Techno.*, vol. 56, no. 3, pp. 244–251, 2016.

[15] G. M. Li B. Zeng, W. Liao, X. Li, and L. Gao, "A new AGV scheduling algorithm based on harmony search for material transfer in a real-world manufacturing system," *Adv. Mech. Eng.*, vol. 10, no. 3, pp. 1–13, 2018.

[16] M. Mousavi, H. J. Yap, S. N. Musa, F. Tahriri, and S. Z. M. Dawal, "Multi-objective AGV scheduling in an FMS using a hybrid of genetic algorithm and particle swarm optimization," *PLoS One*, vol. 12, no. 3, pp. 1–24, 2017.

[17] F. Hamed and M. Nezam, "An optimal path in a Bi-criteria AGV-based flexible job shop manufacturing system having uncertain parameters," *Int. J. Ind. Syst. Eng.*, vol. 13, no. 1, pp. 27–55, 2010.

[18] H. Fazlollahtabara and M. Saidi-Mehrabad, "Optima path in an intelligent AGV based manufacturing system," *Transp. Lett.*, vol. 7, no. 4, pp. 219–228, 2015.

[19] E. Gawrilow, M. Klimm, R. H. Möhring, and B. Stenzel, "Conflict-free vehicle routing," *EURO J. Transp. Logistics*, vol. 1, nos. 1–2, pp. 87–111, Jun. 2012.

[20] C. Oboth, R. Batta, and M. Karwan, "Dynamic conflict-free routing of automated guided vehicles," *Int. J. Prod. Res.*, vol. 37, no. 9, pp. 2003–2030, Jun. 1999.

[21] Y. Xue and J.-Q. Sun, "Solving the path planning problem in mobile robotics with the multi-objective evolutionary algorithm," *Appl. Sci.*, vol. 8, no. 9, p. 1425, Aug. 2018.

[22] X. Ma, Y. Bian, and F. Gao, "An improved shuffled frog leaping algorithm for multiload AGV dispatching in automated container terminals," *Math. Problems Eng.*, vol. 2020, pp. 1–13, Jan. 2020.

[23] T. Le-Anh and M. B. M. De Koster, "A review of design and control of automated guided vehicle systems," *Eur. J. Oper. Res.*, vol. 171, no. 1, pp. 1–23, May 2006.

[24] C. Sahin, M. Demirtas, R. Erol, A. Baykasoglu, and V. Kaplanoglu, "A multi-agent based approach to dynamic scheduling with flexible processing capabilities," *J. Intell. Manuf.*, vol. 28, no. 8, pp. 1827–1845, Dec. 2017.

[25] H. S. A. Eskandar and A. H. M. Bahreininejad, "Water cycle algorithm—A novel metaheuristic optimization method for solving constrained engineering optimization problems," *Comput. Struct.*, vol. 110, pp. 151–166, Nov. 2012.

[26] L. Z. Du, S. Ke, Z. Wang, J. Tao, L. Yu, and H. Li, "Research on multi-load AGV path planning of weaving workshop based on time priority," *Math. Biosci. Eng.*, vol. 16, no. 4, pp. 2277–2292, 2019, doi: 10.3934/mbe.2019113.

[27] P. Udhayakumar and S. Kumanan, "Task scheduling of AGV in FMS using non-traditional optimization techniques," *Int. J. Simul. Model.*, vol. 9, no. 1, pp. 28–39, Mar. 2010.

[28] X. Xiao, Y. Pan, L. Lv, and Y. Shi, "Scheduling multi–mode resource–constrained tasks of automated guided vehicles with an improved particle swarm optimization algorithm," *IET Collaborative Intell. Manuf.*, vol. 3, no. 2, pp. 93–104, Jun. 2021.

PARKAVI KRISHNAMOORTHY received the B.E. degree in information technology and the M.E. and Ph.D. degrees from the College of Engineering Guindy, Anna University, in 2004, 2008, and 2011, respectively. She has more than 18 years of teaching and research experience. She held various academic and research positions. She is currently working as an Assistant Professor-Senior Grade with VIT University Chennai. She has authored many research articles in peer-reviewed journals and a recognized reviewer of reputed journals. Her research interests include interdisciplinary areas, which include cyber security, computer vision, cloud computing, bio-inspired computing, and machine learning.

N. SATHEESH received the B.E. degree in electronics and communication engineering from the Sri Balaji Chockalingam Engineering College, Arani, Tamil Nadu, in 2004, the M.E. degree in computer science and engineering from the Faculty of Engineering and Technology, Annamalai University, Chidambaram, Tamil Nadu, in 2008, and the Ph.D. degree in computer science and engineering from the Karpagam Academy of Higher Education, Coimbatore, Tamil Nadu, in 2018. Since May 2019, he has been working as a Professor with the Department of Computer Science and Engineering, St. Marti's Engineering College, Dhulapally, Secunderabad. He blends his background in electronics and communication engineering and computer science and engineering with high research insight. As part of the research work, more than ten Scopus and six SCI, Springer, Elsevier Journals, and ten Patents have been published. He received the Teaching and Research Excellence National Award, in 2020; the Global Eminent Teachers Award, in 2021; and the Dr. Sarvepalli Radhakrishnan Life-Time Achievement National Award, in 2021. He organized events like International Conference, Seminar, Workshop, STTP, and a Guest Lecture.

D. SUDHA received the B.E. degree in computer science from Anna University, Chennai, in 2010, the M.E. degree in computer science and engineering from Anna University, in 2012, and the Ph.D. degree from VIT University, Chennai, in 2021. She has been working as an Associate Professor with the Department of Computer Science and Engineering, Panimalar Engineering College, Poonamallee, Chennai, since 2021. She has ten years of teaching and research experience in image processing, artificial intelligence areas. Her current research interest includes automated predictions in image processing.

SUDHAKAR SENGAN (Member, IEEE) received the M.E. degree from the Faculty of Computer Science and Engineering, Anna University, Chennai, Tamil Nadu, India, in 2007, and the Ph.D. degree in information and communication engineering, Anna University. He has 20 years of experience in teaching/research/industry. He is currently working as a Professor and the Director of international relations with the Department of Computer Science and Engineering, PSN College of Engineering and Technology (Autonomous), Tirunelveli, Tamil Nadu, India. He guided more than 100 Projects for UG and PG students in engineering streams. He is also the Recognized Research Supervisor with Anna University, under Information and Communication Engineering Faculty. He has published papers in 140 international journals, 20 international conferences, and ten national conferences. He has published three text books for Anna University, Chennai Syllabus. He has filed 20 Indian and three international patents in various fields of interest. His research interests include security, MANET, the IoT, cloud computing, and machine learning. He is a member of professional bodies such as MISTE, MIAENG, MIACSIT, MICST, MIE, and MIEDRC. He received the Award of Honorary Doctorate (Doctor of Letters-D.Litt.) from International Economics University, SAARC Countries in Education and Students Empowerment, in April 2017.

MESHAL ALHARBI received the M.Sc. degree in computer science from Wayne State University, USA, in 2014, and the Ph.D. degree in computer science from Durham University, U.K., in 2020. He has ten years of experience in teaching/research/industry. He is currently an Assistant Professor of artificial intelligence with the Department of Computer Science, Prince Sattam Bin Abdulaziz University, Saudi Arabia. His research interests include artificial intelligence applications and algorithms, agent-based modeling and simulation applications, disaster/emergency management and resilience, optimization applications, and machine learning.

DENIS A. PUSTOKHIN received the Ph.D. degree in logistics and supply chain management from the State University of Management, Moscow, Russia. He is currently an Associate Professor with the State University of Management. He has published over 40 conferences and journal papers. His research interests include enterprise logistics planning, artificial intelligence, big data, the Internet of Things, and reverse logistics network design.

IRINA V. PUSTOKHINA received the M.B.A. and Ph.D. degrees in logistics and supply chain management from the State University of Management, Moscow, Russia. She is currently an Associate Professor with the Plekhanov Russian University of Economics, Moscow. She has published over 40 conferences and journal papers. Her research interests include supply chain management, regional logistics development, sustainable urban development, city logistics, intelligent logistics systems, big data technology and applications, information management, and the Internet of Things.

ROY SETIAWAN received the Bachelor of Informatics Engineering degree from Petra Christian University Informatics Engineering Study Program, in 2004, the Masters of Management Science (M.S.M.) degree from Airlangga University, and the Ph.D. degree (Hons.) in management sciences from Airlangga University, Surabaya, in 2021, with the Domestic Postgraduate Scholarship from the Ministry of Research, Technology and Higher Education of the Republic of Indonesia. In addition to formal education, he also completed certification in the field of Human Resources and Behavior, namely Certified Professional Human Resources (CPHR), Certified Behavioral Analyst (CBA), Certified Behavior Consultant (CBC), and Certified International Trainer (CIT). He is currently serving as a Permanent Lecturer with the Business Management Program, School of Business and Management, Petra Christian University Surabaya, Indonesia, teaching at the Master of Management Study Program. His writings have been published in many reputable international journals ranging from indexed journals Scopus to *Web of Science (WOS)* with topics around management, leadership, organizational behavior, and technology, as well as being a reviewer in Scopus journals. He is currently a member of the Indonesian Doctor of Economics (IDEI) and the Accounting, Management, and Economics Forum (FAME). He has served as the Chair of the Business Management Program and the Head of the Leadership Laboratory. While completing his undergraduate degree, he was also active as the Chair of the Informatics Engineering Student Association for 2002–2003 and once entrusted with being the Chair of the Student Executive Board Petra Christian University for 2003–2004. In addition, he also served as the Coordinator of the Communication Forum for Indonesian Christian Higher Education Student Organizations. His leadership experience became even more skilled when he served as the Chair of the Surabaya City Youth and Children Department Regional Board for the 2009–2017 period in a non-profit organization.