Int J Adv Manuf Technol (1998) 14:428–440

The International Journal of Advanced Manufacturing Technology

AGV Schedule Integrated with Production in Flexible Manufacturing Systems

N. Jawahar*, P. Aravindan†, S. G. Ponnambalam‡ and R. K. Suresh†

*Department of Mechanical Engineering, Triagarajar College of Engineering, Madurai, India, ‡Department of Mechanical Engineering, PSG College of Technology, Coimbatore, India; †Regional Engineering College, Trichirapalli, India; and †Amrita Institute of Technology and Science, Coimbatore, India

Flexible manufacturing systems (FMS) comprise, automated machine tools, automated material handling, and automated storage and automated retrieval systems (AS/RS) as essential components. Effective sequencing and scheduling of the material handling systems (MHS) can have a major impact on the productivity of the manufacturing system. The material handling cannot be neglected while scheduling the production tasks. It is necessary to take into account the interaction between machines, material handling systems and computer. In this context, this paper attempts to link the operation of automated guided vehicles (AGV) with the production schedule and suggests a heuristic algorithm that employs vehicle dispatching rules (vdr) for conflict resolution. The vdrs considered in this paper are: shortest operation time (SPT), longest operation time (LPT), longest travel time (LTT) and shortest travel time (STT). The performance of the vdrs in the proposed heuristic is compared with makespan criteria. The results show that the STT provides the best solutions compared to other vdrs.

Keywords: AGV; Flexible manufacturing system; Heuristics; Scheduling

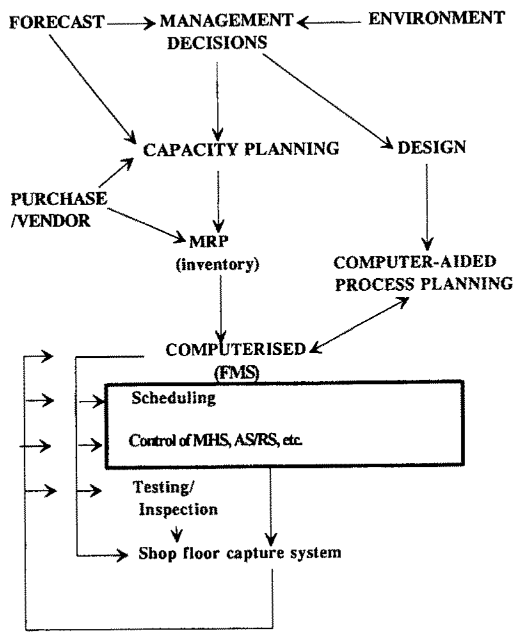

Fig. 1. Production planning and control in CIM.

1. Introduction

A flexible manufacturing system (FMS) is a computer-controlled configuration of various kinds of processing stations, automated material handling systems, and automated storage/retrieval systems (AS/RS). The strategic role of a material handling system (MHS) in an FMS as given by Bedworth and Bailey [1] is given in Fig. 1. The main function of an MHS is to supply the correct materials at the correct locations, and at the correct time. The cost of material handling is a significant portion of total cost of production. Eynan and Rosenblatt [2] indicated that the material handling cost is about 30% of the total production cost. This makes the subject of material handling increasingly important. Effective sequencing and scheduling of the MHS can have a major impact on the productivity of the manufacturing system [3]. This is especially true in the case where material handling times are comparable with machine processing times and cannot be neglected while scheduling the production tasks [3]. The integration of the machines into a system, achieved by automated material handling and by the overall computer control, can result in manufacturing systems characterised by flexibility, high productivity and low cost per unit produced. It is necessary to take into account the interaction between machines, material handling systems and computer [4]. Hence, the role of an MHS in an FMS is important and further, the operation of an MHS requires coordinated effort, as it has to conform to the production

Correspondence and offprint requests to: Dr S. G. Ponnambalam, Department of Mechanical Engineering, PSG College of Technology, PO Box 1611, Peelamedu, Coimbatore 641 004, India.

schedule. In this context, the purpose of this paper is to link the operation of an MHS with the production schedule.

This paper considers an FMS that is required to process various types of job loaded at discrete points of time at different processing stations. Each job requires a prescribed sequence of operations that specifies the order in which the operations are to be performed. The production demands a flexible MHS to move the parts to various processing stations during a production run. Automated guided vehicles (AGV) are used in many material handling situations involving large, heavy loads over flexible routes. AGVs are the most flexible of the floor cart systems and follow electromagnetic impulses transmitted from a guide wire embedded in the plant floor. Two advantages of AGV systems, in a FMS, are that they can accommodate many work stations and can communicate with a computer-controlled system on a real-time basis [5]. AGV systems continue to play a significant role in low- to medium-flow manufacturing operations, including FMS and other applications. The relatively inexpensive guide-path which does not interfere with other material flow systems in the facility, coupled with the high degree of flexibility and control offered in vehicle routeing, has made AGV systems a proven and viable handling technology [6]. An AGV is a driverless vehicle that accomplishes the material handling tasks flexibly and so is considered appropriate for an FMS environment [7]. For this reason an AGV system is selected for the material handling tasks in the FMS modelled.

Uluso and Bilge [8] pointed out that the majority of the reported work that deals with the subject of material-handling-system scheduling is set out as a comparison of various vehicle dispatching rules (*vdrs*) in relation to a prespecified schedule and for a particular layout. They further indicated that the coordination and integration with machine scheduling during the scheduling phase of the FMS has not received much attention. Their paper is an attempt to make the scheduling of AGVs an integral part of the overall scheduling activity in an FMS environment. They proposed an iterative solution procedure to generate schedules for machines and AGVs simultaneously. Their methodology is based on the following components:

1. An algorithm that generates machine schedules.
2. An algorithm that finds a feasible solution to the vehicle scheduling problem given the machine schedule.
3. An iterative structure that links the two and facilitates the search for a good solution.

They employed Giffler and Thompson's active schedule algorithm [9] and a non-delay schedule algorithm [10] using different priority dispatching rules for machine-schedule generation. They applied a sliding time window (STW heuristic) concept to provide a feasible vehicle schedule. Since the iterative and heuristic procedure requires more CPU time, the authors themselves doubted that the iterative procedure could be adapted for real-time dynamic scheduling. This paper attempts to link the operation of an AGV with the production schedule. A heuristic algorithm that employs vehicle dispatching rules for conflict resolution is suggested. The algorithm provides the chronological events of the entire operation of the system, which is addressed as an integrated schedule (IS), that enables the derivation of an AGV schedule (AGVS) integrated with a modified production schedule (MPS). The aim of the present work is to find an optimal IS that provides a feasible MPS and AGVS.

The paper is organised as follows. In Section 2, the related literature that helped to formulate the model and algorithm is discussed. Section 3 addresses the problem environment. The proposed heuristic algorithm is addressed in Section 4. An illustration is presented in Section 5. In Section 6, the performance of a few *vdrs* in the proposed heuristic algorithm is discussed. The conclusions and future research directions are presented in the final section.

2. Design and Operation of AGV Systems

The performance of an AGV must be sufficiently good to justify the high cost involved. The method of operating an AGV affects its performance. The efficient use of an AGV system depends on design (configuration and guide-path system) and operating strategies (routeing practices and dispatching policies).

2.1 Design of AGV Systems

2.1.1 Configuration of AGV Systems

Most of the AGV systems employ a traditional (conventional) configuration. A traditional system refers to the case where a fleet of vehicles serves a set of stations which are defined by the location of their pick-up and deposit (P/D) points. In such systems, any vehicle is allowed to serve any P/D point and all the vehicles operate over the same guide path. Since each vehicle has access to all stations, conventional AGV systems generally require extensive control systems for vehicle dispatching, vehicle routeing and traffic management. The control systems are complex. Bozer and Srinivasan [6] proposed tandem configurations that facilitate the use of distributed processing. They indicated that tandem configurations offer simplicity and flexibility, facilitate future expansion, require two or more vehicles to fulfil one handling task, require additional space (owing to non-overlapping loops and additional P/D points to interface adjacent loops), require balanced workloads among the routes to avoid bottleneck loops, and generate the load route problem. The tandem configuration is an application of the "divide and conquer" principle to AGV systems. It is based on partitioning all the stations into non-overlapping, single-vehicle closed loops with additional P/D points provided as an interface between adjacent loops. Bozer and Srinivasan [11] developed an analytical model of a single vehicle operating in a closed loop that represents the individual loop in a tandem AGV system and analysed the throughput capacity. The AGV systems can also be classified as "pick and drop" and "stop and go". In "pick and drop" systems, the loads are picked at certain points and delivered through the AGV to their destination points. The AGV in "stop and go" systems functions as a platform on which a job remains until all processing operations are completed [11].

2.1.2 Guide-path System of AGV

Generally, the AGV-based MHS follows either a unidirectional guide-path system or a bidirectional guide-path system [7]. In the unidirectional guide-path system, the AGV is permitted to move in one direction only. In contrast to a unidirectional system, the AGV can change its direction and move in both directions in a bidirectional system. The main reason for the use of unidirectional networks is their simplicity in design and control. The bidirectional systems have to be controlled to prevent vehicle collisions and congestion delays. Egbelu and Tanchoco [12] analysed the production potential of facilities using a bidirectional network and a unidirectional network. They demonstrated through simulation that the bidirectional networks outperformed unidirectional networks in every performance measure. Spinelli [13] found substantial savings and increases in vehicle utilisation using a bidirectional network.

2.2 Operating Strategies of AGV

2.2.1 Routeing Practices

An AGV performs two kinds of trip; a loaded trip and an empty trip. At the end of each loaded trip, the AGV is assigned another loaded trip and routed directly for the next task. This routeing practice is called *direct routeing* [8]. If the assigned trip is to be initiated from a different location, then an empty trip is inevitable even with direct routeing practice. The AGV waits at the destination that it has been directed to. Another routeing practice uses pick and place stations [14]. The trip is initiated from a common loading pick and deposit station. This practice is called *routeing via load/unload station.* This leads to an empty move after each loaded trip. The location of the pick and deposit station influences the performance of the AGV. Uluso and Bilge [8] demonstrated that direct routeing is more efficient than routeing via load/unload station and is especially efficient when the ratio of average travel time to average process time for the problem is greater than 0.4.

2.2.2 Vehicle Dispatching Rules (vdr)

Srinivasan et al. [15] indicated some *vdr*s that are discussed in the literature and they are:

FCFS	first-come-first-served
MFCFS	modified-first-come-first-served
FEFS	first-encountered-first-served
MOQS	maximum-outgoing-queue-size-first
STTF	shortest-travel-time (distance)-first
MODFCFS	MODified-first-come-first-served (another modification)
CLF	closest-load-first
FLF	furthest-load-first
LIV*	longest-idle-vehicle
NV*	nearest-vehicle
RV*	random vehicle
	(*Note: Applicable when number of vehicles is greater than one)

They pointed out that the performance of the analytical models depends on the *vdr* adopted for the system and it is difficult to generalise relationships between the models and the *vdr*s.

In this paper, the FMS is modelled with an AGV that transfers the parts within the system. The AGV system is designed with an AGV that moves along a single closed loop guide path and uses a "pick and drop" system. This resembles Bozer and Srinivasan's single loop of tandem configuration model [11]. the AGV is capable of moving a unit load from one node to another node and is equipped with automatic loading and unloading. Since the bidirectional network outperforms the unidirectional networks in every performance measure [12,13], a bidirectional network is selected for AGV control. Direct routeing practice that avoids unnecessary empty moves between each loaded trip to the load/unload station is used to direct the AGV according to the pre-planned production schedule. The performance of the scheduling methodologies is generally compared with makespan related objectives or due date related objectives [16]. The objective depends upon the operating policy of the firm. Since the FMS belongs to a high investment category, the utilisation function is the primary concern. In this aspect, the objective criterion of makespan time is considered as the performance measure.

3. Problem Description

This section describes the problem with the details of the configuration and the operating environment of the FMS considered.

3.1 Configuration and Operating Environment of FMS

3.1.1 FMS Descriptions

The configuration of the FMS considered is:

1. The overall system comprises (Fig. 2) several computer numerical controlled processors (Work Cell), an automatic

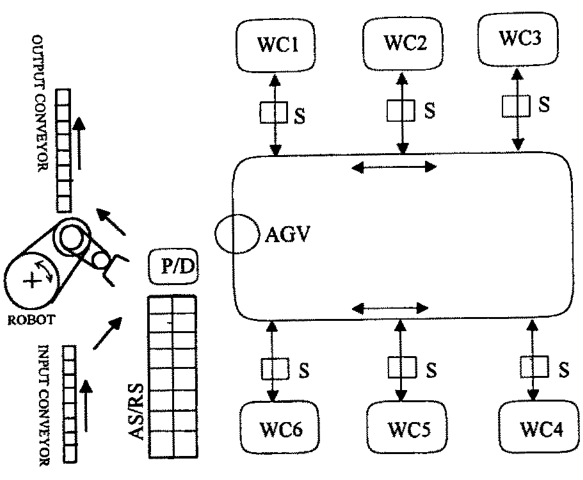

Fig. 2. Configuration of a flexible manufacturing system. WC,work cell; AGV, automated guided vehicle; P/D, pick and deposit; AS/RS, automated storage/retrieval system; S, shuttle.

guided vehicle (AGV), part carrying conveyors (input and output), a robot, and an automated storage and retrieval system (AS/RS). A host computer links and controls the operations associated with them.

2. The number of work cells (WC) considered in the system is 6. A WC is either a machining centre or an assembly machine or an inspection station. Each WC is provided with independent tool magazines, automated part loading and unloading arrangements, part program controller, automatic tool changer (ATC) and buffer storage.
3. The robot transfers the components from AS/RS to output conveyor and from input conveyor to AS/RS.
4. The AGV transfers materials (raw material, WIP and finished product) between WCs, WC and AS/RS, and load/unload station and WC. The design along with the operating strategies of the AGV system is addressed in Section 2. This handles the part flow within the system.
5. A sufficiently large number of general purpose pallets and fixtures is available.
6. The AS/RS stores the raw materials necessary for the parts to be processed as well as the work in progress (WIP) inventories.

3.1.2 Assumptions

1. There are n jobs to be processed in one or more of m facilities or WCs during a certain planning horizon. The raw materials of all the jobs are stored in the AS/RS before the start of the planning horizon.
2. Each job once started must be completed (no pre-emptive priorities).
3. The operation sequences of all jobs along with alternative WC choices for each operation are known. For each operation, a particular WC is selected based on the availability of tools and on the economical aspects. This consideration permits the use of only one WC for one operation and ensures the availability of the tools necessary at the WC at any instant.
4. The maximum number of operations associated with each job is equal to the number of WCs in the system.
5. A revisit for another operation on the same WC on a job is not allowed.
6. The operation time of a job at each facility includes the loading, unloading, tool changeover and set-up times (both tool and workpiece) along with processing time (i.e. the time between the part being picked and returned to the local buffer).

$$T_{ij} = Q * [L_{ij} + \sum_{y=1}^{y=Y_{(ij)}} (M_{y(ij)} + WS_{y(ij)} + TS_{y(ij)}) + UL_{ij}]$$

Where

i = job number

j = work cell (WC) number

L_{ij} = loading time of job i at WCj

$M_{y(ij)}$ = time for machining an operation y of job i at WCj

Q = batch size ($Q \geq 1$)

T_{ij} = processing time of job i on machine j

$TS_{y(ij)}$ = tool changeover time between each machining activity for job i at WCj

$UL_{y(ij)}$ = unloading time of job i at WCj

$WS_{y(ij)}$ = set-up time between each machining activity for job i at WCj

$Y_{(ij)}$ = number of machining operations involved with job i on WCj

7. The transportation time (TT_{ij}) depends upon the locations of WCs and AS/RS (layout of the system), and mode of operation and speed of the AGV. The layout of the system considered is shown in Fig. 2. The distance between the neighbouring elements of the system is the same and is equal to one unit length. The AGV moves at a constant speed of one unit length per unit time in a bidirectional mode. The distance and the transportation time (that includes loading and unloading time) thus arrived at for the layout considered is given in Table 1.
8. The local buffers store the WIP inventories.
9. Breakdowns are neglected.

3.2 Problem Definition

The input to the problem is an optimal production schedule that excludes the transfer activities of the FMS (i.e. the production schedule obtained with zero transportation time). This is referred to here as the "production schedule". It is the timetable of all operations associated with the n jobs (that require processing during the planning period under consideration) in the WCs of the manufacturing model described in Section 3.1 and is the one that excludes the transportation times. The production schedule is represented as a matrix with four elements. It provides the starting and finishing times of every operation of all jobs along with the WC number at which the operation is performed. The four elements of the production schedule matrix are:

Element 1: Indicates the job number i.

Element 2: Indicates the WCj.

Element 3: Indicates the start time (i.e. elapsed time at the start of operation) of the operation that is performed by WCj on job i.

Element 4: Indicates the finish time (i.e. elapsed time at the end of operation) of job i on WCj.

The inclusion of transfer activities shifts the given production schedule to the right which facilitates the AGV movements and modifies the production schedule. The MPS integrates the AGV movements. The combined schedule of MPS and AGVS called IS provides the chronological events of all the operations of the system. The aim of the present work is to find an optimal IS which provides an active feasible MPS and AGVS. The makespan time is considered as the performance criterion and so the objective function is "minimisation of makespan time". If the deviation of MPS from the given optimal pro-

Table 1. Distance/transportation time (TT_{jj}) matrix.

To \ From	AS/RS(0)	WC1 (1)	WC2 (2)	WC3 (3)	WC4 (4)	WC5 (5)	WC6 (6)
AS/RS (0)	0	1	2	3	3	2	1
WC1 (1)	1	0	1	2	3	3	2
WC2 (2)	2	1	0	1	2	3	3
WC3 (3)	3	2	1	0	1	2	3
WC4 (4)	3	3	2	1	0	1	2
WC5 (5)	2	3	3	2	1	0	1
WC6 (6)	1	2	3	3	2	1	0

duction schedule is a minimum, then the schedules are optimal for the makespan criterion. In this context, the problem is defined as follows:

> Derivation from the original production schedule of an optimal integrated schedule for the makespan criterion, which gives a modified production schedule that integrates the AGV moves (AGV schedule) with minimum right shift.

4. Proposed Methodology

The proposed methodology, which derives IS, MPS and AGVS from a production schedule, is addressed in this section. This involves three stages as follows:

Stage 1. Data Modulation Module

This module converts the given production schedule matrix to suit the programming environment. The two steps involved in this module are described below.

Step 1. Conversion of Production Schedule Matrix to Preschedule Matrix

The given production schedule matrix is converted to a preschedule matrix that comprises a set of task vectors, the size of which depends on the number of jobs n and the number of WCs m. There are five elements in a task vector that contain the following information:

Element 1: Indicates the job i.

Element 2: Indicates the operation sequence identifier k of job i.

Element 3: Indicates the starting time of operation k of job i on WCj.

Element 4: Indicates the operation time T_{ij} of job i on WCj.

Element 5: Indicates the WC number j at which the operation is performed.

For each job i, the task vectors thus generated are $m + 2$. They are indicated below:

1. There is one task vector for each operation associated with job i. The number of such vectors is equal to the number of operations N_k_i (Note: $N\_k_i \leq m$, i.e. assumption 4) to be performed for job i.
2. The $(N\_k_i + 1)$ task vector indicates the completion of all operations and the return to AS/RS. The elements 3, 4 and 5 are indicated with zero.
3. Then $(m - N\_k_i)$ task vectors are added to the above to maintain uniformity in representing the data. The values of elements 3, 4 and 5 are indicted with zero.
4. The $(m + 2)$ vector is the indicator of the end of the task vectors associated with job i and the values of its elements 3 and 5 are zero and -1 is added to element 4.

Step 2. Conversion of Preschedule Matrix to Task Schedule Matrix

The preschedule matrix is modified by adding one more element 6 *opcount* that represents the completion status of the tasks. If the *opcount* value is -1, then the task is considered as unassigned. The *opcount* changes to $+1$ after it is assigned.

Element 6: Indicates the assignment of the task.

A row in a task schedule matrix thus comprises six elements. Initially element 6 is assigned a value -1. This task schedule matrix becomes the input for the proposed heuristic algorithm.

Stage 2. Initialisation Module

A set of variables that comprises busy status variables and location variables represents the status of the WCs, the jobs, and the AGV. The busy status variables (*busy_sts*) indicate the availability of WCs, the jobs and AGV at any instant of the program run. The variable *busy_sts(j)*represents the status of WCj and the number of such variables is equal to the number of WCs. The variable *busy_sts(i)* represents the status of job i and the number of such variables is equal to the number of jobs. The variable *busy_sts(AGV)* represents the status of the AGV. The location status variables (*loc_sts*) indicate the location of jobs and the AGV during the different stages of schedule revision. The variable *loc(i)* indicates the location of job i. The location of AGV is identified by another variable *loc(AGV)*. Whenever a WC or a job or an AGV is assigned an operation, then its respective *busy_sts* variable is set with the value that indicates the time from which it is available for the next activity. The *loc_sts variables* of jobs and AGV change according to their current locations. The current time is set as 0. Since all the jobs and WCs, and AGVs are available initially, the *busy_sts* variables are initialised to 0 which indicates the availability of all WCs, jobs and AGVs.

All the *loc_sts* variables are also set to 0 to indicate that the AGV and the jobs are initially as AS/RS.

Stage 3. Schedule Generation Module

This module employs a heuristic algorithm to generate the following schedules:

1. IS shows the chronological events of the entire operations.
2. MPS shows the revised production schedule.
3. AGVS provides the movements of the AGV.

The various steps that are involved in the schedule generation heuristic are given below.

Step 1. Initialisation of Task List Queue of AGV (AGV_task_list)

The queue of tasks that wait for transportation before the AGV is called the *AGV_task_list*. Since the raw materials of all jobs are available in AS/RS at the start of the planning period and require processing at least in one WC, all the jobs are waiting for the transportation service on an AGV. Hence, select the task vectors associated with the first operation (i.e. $k = 1$) of all the jobs from the task schedule matrix and store them in *AGV_task_list*.

Step 2. Selection of a Task for AGV Move and Assignment

A task is selected from the *AGV_task_list* based on the status of WCs, jobs and AGV. The selected task vector assigns the associated tasks (i.e. AGV move and the operation in WC). The various steps of the selection procedure are as follows:

1. Find the list of task vectors that are the candidates for an AGV move (*AGV_task_list*1) with the following guidelines:
   (*a*) Remove the task vectors that require processing on WCs that are engaged with some other operation. The status of WC*j* is identified by checking the *busy_sts*(*j*) that indicates the time from which the WC*j* is available with the *busy_sts*(*AGV*) that indicates the current time in the program run. The WC*j* is busy if *busy_sts*(*j*) is greater than *busy_sts*(*AGV*).
   (*b*) Remove the task vectors of the jobs that are busy. A job is busy when its previous operation is not yet over. The status of the job *i* is found by comparing the *busy_sts*(*i*) with *busy_sts*(*AGV*).
2. Find the task vectors that have the least starting time (element 3) from the *AGV_task_list*1 and label as *AGV_task_list*2. Count the number of task vectors in *AGV_task_list*2 and store as *task_count*.
3. Determine the task vector for assignment. If *task_count* is one, then the task vector in *AGV_task_list*2 becomes the task vector for assignment.

If *task_count* is greater than one, then a *vdr* resolves the conflict between the task vectors in *AGV_task_list*2 and selects a task vector.

The *vdr*s used are given in Table 2.

Step 3. Append the IS, MPS, AGVS Data Files

Integrated Schedule. A row of six elements in the integrated schedule data file represents every move of the AGV. The six elements are:

Element 1: AGV movement index.
Element 2: Indicates that job *i* is loaded on the AGV (the value 0 indicates the empty move of AGV).
Element 3: Indicates the operation sequence identifier *k* of job *i* (the value 0 indicates the empty move of AGV).
Element 4: Indicates the earliest starting of AGV from the WC*j* that is indicated in the 6th element for the next move. It also indicates the delivery time of job *i* at WC*j* and the start time of operation *k* of job *i*.
Element 5: Finish time of operation *k* of job *i* on the WC*j*.
Element 6: Indicates the WC number *j* at which the operation *k* of job *i* is performed. This is the current location of the AGV and job *i*. The AGV starts moving from this location for the next move.

The selected task vector supplements the above elements with the information that is stored in the task schedule matrix, distance/time matrix, and *busy_sts*( )and *loc*( ) variables.

Modified Production Schedule. Whenever a task vector is selected for an AGV move and assigned, the start and finish

Table 2. Vehicle dispatching rules.

Rule	Queue discipline	Explanation
Shortest oPeration Time (SPT)	Ranked in the ascending order of operation times (T_{ij})	Selects a task vector that has minimum operation time from *AGV_task_list2*
Longest oPeration Time(LPT)	Ranked in the descending order of operation times (T_{ij})	Selects the task that has maximum operation time from *AGV_task_list2*
Furthest Travel Time (FTT)	Ranked in the descending order of travel times (TT_{jj})	Selects a task vector that requires maximum travel time from *AGV_task_list2*
Shortest Travel Time (STT)	Ranked in the ascending order of travel times (TT_{jj})	Selects the task that requires minimum travel time from *AGV_task_list2*

Table 3. Job data.

	i	1		2		3		4		5		6		7		8		9		10
k	*j*	T_{ij}	*j*	T_{ij}	*j*	T_{ij}	*j*	T_{ij}	*j*	T_{ij}	*j*	T_{ij}	*j*	T_{ij}	*j*	T_{ij}	*j*	T_{ij}	*j*	T_{ij}
1	2	24	3	25	2	20	1	25	2	30	2	20	3	15	1	40	2	12	3	35
2	1	16	2	30	1	25	3	35	1	20	1	20	1	15	2	10	4	23	2	45
3	3	20	1	40	3	15	2	45	3	40	3	30	4	20	6	15	6	15	5	30
4	5	10	6	15	4	10	5	15	4	10	6	15	6	10	5	25	0	0	4	10
5	6	10	0	0	5	5	6	20	6	10	4	10	0	0	0	0	0	0	0	0
6	0	0	0	0	6	5	4	10	0	0	5	5	0	0	0	0	0	0	0	0
D_i		280		360		160		750		660		450		240		270		100		360

Table 4. Production schedule matrix.

i	*j*	st_{ij}	ft_{ij}
1	2	50	74
1	1	90	106
1	3	106	126
1	5	126	136
1	6	151	161
2	3	50	75
2	2	119	149
2	1	149	189
2	6	189	204
3	2	12	32
3	1	65	90
3	3	90	105
3	4	105	115
3	5	115	120
3	6	120	125
4	1	40	65
4	3	126	161
4	2	199	244
4	5	244	259
4	6	259	279
4	4	289	299
5	2	169	199
5	1	209	229
5	3	239	279
5	4	279	289
5	6	289	299
6	2	149	169
6	1	189	209
6	3	209	239
6	6	239	254
6	4	254	264
6	5	264	269
7	3	0	15
7	1	106	121
7	4	121	141
7	6	141	151
8	1	0	40
8	2	40	50
8	6	50	65
8	5	65	90
9	2	0	12
9	4	12	35
9	6	35	50
10	3	15	50
10	2	74	119
10	4	136	166
10	4	166	176

times of that operation are inferred from the IS data file and the modified production schedule is appended. The four elements of the MPS are described below.

Element 1: Job number *i* (element 2 of IS).

Element 2: WC number *j* (element 6 of IS).

Element 3: Starting time of operation *k* of job *i* in WC*j* (element 4 of IS).

Element 4: Finishing time of operation *k* of job *i* in WC*j* (element 5 of IS).

AGV Schedule. The AGVS is represented by five elements that describe the activities of the AGV.

Element 1: Indicates that job *i* is loaded on the AGV (element 2 in IS provides the information).

Element 2: Indicates the starting location of the AGV (element 6 of previously appended row in IS provides the information).

Element 3: Indicates the destination of the AGV (element 6 of the recently appended row in IS provides the information).

Element 4: Indicates the starting time of the AGV from the location that is indicated in element 2 (element 4 of the previously appended row in IS provides the information).

The above pieces of information are obtained by manipulating the IS data file.

Step 4. Update the Status of WC, Job, AGV and AGV_task_list

The status of job *i* that is selected and assigned in WC*j*, WC*j* and AGV are updated with the values derived from the IS file. The values correspond to the values in the IS data file for the task selected and assigned.

busy_sts(i): element 5 in the last row of IS data file.
busy_sts(j): element 5 in the last row of IS data file.
busy_sts(AGV): element 4 in the last row of IS data file.
loc_sts(AGV): element 6 in the last row of IS data file.
loc_sts(i): element 6 in the last row of IS data file.

The task vector of operation *k* of job *i* in WC*j* assigned for the AGV move is removed from the AGV_task_list and the

Table 5. Preschedule (elements 1 to 5) and task schedule matrices (elements 1 to 6).

Element 1 Job i	Element 2 Operation k	Element 3 Start time	Element 4 Operation time T_{ij}	Element 5 WC/j	Element 6	Element 1 Job i	Element 2 Operation k	Element 3 Start time	Element 4 Operation time T_{ij}	Element 5 WC/j	Element 6
1	1	50	24	2	−1	6	1	149	20	2	−1
1	2	90	16	1	−1	6	2	189	20	1	−1
1	3	106	20	3	−1	6	3	209	30	3	−1
1	4	126	10	5	−1	6	4	239	15	6	−1
1	5	151	10	6	−1	6	5	254	10	4	−1
1	6	0	0	0	−1	6	6	264	5	5	−1
1	7	0	0	0	−1	6	7	0	0	0	−1
1	8	0	−1	0	−1	6	8	0	−1	0	−1
2	1	50	25	3	−1	7	1	0	15	3	−1
2	2	119	30	2	−1	7	2	106	15	1	−1
2	3	149	40	1	−1	7	3	121	20	4	−1
2	4	189	15	6	−1	7	4	141	10	6	−1
2	5	0	0	0	−1	7	5	0	0	0	−1
2	6	0	0	0	−1	7	6	0	0	0	−1
2	7	0	0	0	−1	7	7	0	0	0	−1
2	8	0	−1	0	−1	7	8	0	−1	0	−1
3	1	12	20	2	−1	8	1	0	40	1	−1
3	2	65	25	1	−1	8	2	40	10	2	−1
3	3	90	15	3	−1	8	3	50	15	6	−1
3	4	105	10	4	−1	8	4	65	25	5	−1
3	5	115	5	5	−1	8	5	0	0	0	−1
3	6	120	5	6	−1	8	6	0	0	0	−1
3	7	0	0	0	−1	8	7	0	0	0	−1
3	8	0	−1	0	−1	8	8	0	−1	0	−1
4	1	40	25	1	−1	9	1	0	12	2	−1
4	2	125	35	3	−1	9	2	12	23	4	−1
4	3	199	45	2	−1	9	3	35	15	6	−1
4	4	244	15	5	−1	9	4	0	0	0	−1
4	5	259	20	6	−1	9	5	0	0	0	−1
4	6	289	10	4	−1	9	6	0	0	0	−1
4	7	0	0	0	−1	9	7	0	0	0	−1
4	8	0	−1	0	−1	9	8	0	−1	0	−1
5	1	169	30	2	−1	10	1	15	35	3	−1
5	2	209	20	1	−1	10	2	74	45	2	−1
5	3	239	40	3	−1	10	3	136	30	5	−1
5	4	279	10	4	−1	10	4	166	10	4	−1
5	5	289	10	6	−1	10	5	0	0	0	−1
5	6	0	0	0	−1	10	6	0	0	0	−1
5	7	0	0	0	−1	10	7	0	0	0	−1
5	8	0	−1	0	−1	10	8	0	−1	0	−1

Table 6. List of task vectors in AGV queue *AGV_task_list*.

Element 1 Job i	Element 2 Operation k	Element 3 Start time	Element 4 Operation time T_{ij}	Element 5 WC/j	Element 6
1	1	50	24	2	−1
2	1	50	25	3	−1
3	1	12	20	2	−1
4	1	40	25	1	−1
5	1	169	30	2	−1
6	1	149	20	2	−1
7	1	0	15	3	−1
8	1	0	40	1	−1
9	1	0	12	2	−1
10	1	15	35	3	−1

Table 7. List of contending jobs *AGV_task_list2*.

Element 1 *Job i*	*Element* 2 *Operation k*	*Element* 3 *Start time*	*Element* 4 *Operation time* T_{ij}	Element 5 WC *j*	Element 6
7	1	0	15	3	−1
8	1	0	40	1	−1
9	1	0	12	2	−1

Table 8. Updated list of tasks *AGV_task_list*.

Element 1 Job *i*	Element 2 Operation *k*	Element 3 Start time	Element 4 Operation time T_{ij}	Element 5 WC *j*	Element 6
1	1	50	24	2	−1
2	1	50	25	3	−1
3	1	12	20	2	−1
4	1	40	25	1	−1
5	1	169	30	2	−1
6	1	149	20	2	−1
7	1	0	15	3	−1
8	1	0	40	1	−1
10	1	15	35	3	−1
9	2	12	23	4	−1

6th element of the task schedule matrix corresponding to this task vector is changed to +1 to indicate the assignment of that activity. The task vector of $(k + 1)th$ *operation* of job *i* is added to the *AGV_task_list*.

Step 5. Termination

Steps 2 and 5 are repeated until all the tasks from the task schedule matrix are exhausted. When all the element 6 of the task schedule matrix become +1 this indicates the termination.

5. Example

The proposed methodology is illustrated with an example in this section. The optimal production schedule for the production requirement during a planning horizon (Table 3) is given in Table 4. This becomes the input for the problem under consideration. The preschedule matrix and the task schedule matrix obtained using the data modulation module are given in Table 5.

Stage 2. Initialisation Module

The values of the variables that indicate the status of WCs, jobs, and AGV are initialised with zero.

$$busy\_sts(i) = 0 \quad \forall i$$

$$busy\_sts(j) = 0 \quad \forall j$$

$$busy\_sts(AGV) = 0$$

$$loc\_sts(AGV) = 0$$

$$loc\_sts(i) = 0 \quad \forall i$$

Stage 3. Schedule Generation Module

Step 1. Initialisation of Task List Queue of AGV (AGV_task_list)

All the task vectors associated with operation sequence number one (i.e. $k = 1$) are selected from the task schedule matrix and become the members in the *AGV_task_list* matrix (Table 6).

Step 2. Selection of a Task for AGV Move and Assignment

Since the status of all WCs, jobs and AGV is free, all the tasks of *AGV_task_list* are the eligible candidates for AGV move and *AGV_task_list1* is identical to *AGV_task_list*. The list of eligible tasks in queue *AGV_task_list 1* is the same as the list given in Table 6.

The task vectors that have the least starting time (element 3) from the *AGV_task_list1* are associated with jobs 7, 8 and 9. This provides the *AGV_task_list2* that is given in Table 7.

The number of tasks contending for the services of AGV at time 0 is three and hence *vdrs* resolve the conflict.

SPT rule selects the task that is associated with job 9 (the task with minimum operation time).

LPT rule selects the task that is associated with job 8 (the task with maximum operation time).

FTT rule selects the task that is associated with job 7 (the task with maximum travel distance).

Table 9. Integrated schedule matrix (SPT Vehicle dispatching rule).

Element 1	Element 2	Element 3	Element 4	Element 5	Element 6	Element 1	Element 2	Element 3	Element 4	Element 5	Element 6	Element 1	Element 2	Element 3	Element 4	Element 5	Element 6
0	0	0	0	0		44	0	0	115	5		88	10	4	231	241	4
1	9	1	2	14	2	45	8	5	117	117	0	89	0	0	234	1	
2	0	0	4	0		46	0	0	118	1		90	6	3	236	266	3
3	7	1	7	22	3	47	4	2	120	155	3	91	0	0	241	5	
4	0	0	10	0		48	0	0	123	0		92	1	5	242	252	6
5	8	1	11	51	1	49	8	6	123	123	0	93	0	0	244	4	
6	0	0	13	0		50	0	0	124	0		94	10	5	247	247	0
7	3	1	15	35	2	51	8	7	124	124	0	95	0	0	249	2	
8	9	2	17	40	4	52	0	0	125	0		96	5	2	250	270	1
9	0	0	20	0		53	7	7	125	125	0	97	0	0	251	0	
10	10	1	23	58	3	54	0	0	129	2		98	10	6	251	251	0
11	0	0	41	4		55	3	2	130	155	1	99	0	0	254	3	
12	9	3	43	58	6	56	0	0	136	3		100	4	3	255	300	2
13	0	0	52	3		57	10	2	137	182	2	101	0	0	257	0	
14	7	2	54	69	1	58	0	0	156	1		102	10	7	257	257	0
15	8	2	55	65	2	59	3	3	158	173	3	103	0	0	258	6	
16	0	0	57	0		60	0	0	159	2		104	1	6	259	259	0
17	2	1	60	85	3	61	2	3	160	200	1	105	0	0	260	0	
18	0	0	63	6		62	0	0	174	3		106	1	7	260	260	0
19	9	4	64	64	0	63	3	4	175	185	4	107	0	0	267	3	
20	0	0	66	2		64	0	0	178	1		108	6	4	270	285	6
21	8	3	69	84	6	65	1	3	180	200	3	109	0	0	272	1	
22	0	0	70	0		66	0	0	183	0		110	5	3	274	314	3
23	9	5	70	70	0	67	6	1	185	205	2	111	0	0	286	6	
24	0	0	71	1		68	0	0	187	4		112	6	5	288	298	4
25	7	3	74	94	4	69	3	5	188	193	5	113	0	0	299	4	
26	0	0	77	0		70	0	0	194	2		114	6	6	300	305	5
27	9	6	77	77	0	71	10	3	197	227	5	115	0	0	306	2	
28	1	1	79	103	2	72	3	6	198	203	6	116	4	4	309	324	5
29	0	0	81	0		73	0	0	204	1		117	6	7	311	311	0
30	9	7	81	81	0	74	2	4	206	221	6	118	0	0	315	3	
31	4	1	82	107	1	75	3	7	207	207	0	119	5	4	316	326	4
32	0	0	85	6		76	0	0	209	2		120	0	0	325	5	
33	8	4	86	111	5	77	6	2	210	230	1	121	4	5	326	346	6
34	0	0	95	4		78	0	0	211	0		122	0	0	347	4	
35	7	4	97	107	6	79	5	1	213	243	2	123	5	5	349	359	6
36	0	0	104	3		80	0	0	222	6		124	4	6	351	361	4
37	2	2	105	135	2	81	2	5	223	223	0	125	0	0	360	6	
38	0	0	108	6		82	0	0	224	0		126	5	6	361	361	0
39	7	5	109	109	0	83	2	6	224	224	0	127	0	0	364	4	
40	0	0	111	2		84	0	0	225	0		128	4	7	367	367	0
41	1	2	112	128	1	85	2	7	225	225	0	129	0	0	368	0	
42	0	0	113	0		86	0	0	228	3		130	5	7	368	368	0
43	7	6	113	113	0	87	1	4	230	240	5						

Table 10. Modified production schedule (SPT Vehicle dispatching rule).

job i	WC j	Starting time*	Finishing time
9	2	2	14
7	3	7	22
8	1	11	51
3	2	15	35
9	4	17	40
10	3	23	58
9	6	43	58
7	1	54	69
8	2	55	65
2	3	60	85
8	6	69	84
7	4	74	94
1	2	79	103
4	1	82	107
8	5	86	111
7	6	97	107
2	2	105	135
1	1	112	128
4	3	120	155
3	1	130	155
10	2	137	182
3	3	158	173
2	1	160	200
3	4	175	185
1	3	180	200
6	2	185	205
3	5	188	193
10	5	197	227
3	6	198	203
2	6	206	221
6	1	210	230
5	2	213	243
1	5	230	240
10	4	231	241
6	3	236	266
1	6	242	252
5	1	250	270
4	2	255	300
6	6	270	285
5	3	274	314
6	4	288	298
6	5	300	305
4	5	309	324
5	4	316	326
4	6	326	346
5	6	349	359
4	4	351	361

STT rule selects the task that is associated with job 8 (the task with minimum operation time).

Step 3. Append the IS, MPS, AGVS Data Files

If the selected task corresponds to job 9 (as per *SPT vdr*), then the IS data file is appended with the following values:

Element 1: 1 (AGV move number).

Element 2: 9 (Job number i).

Element 3: 1 (Operation sequence number k).

Element 4: 2 (Arrival time of AGV at WC2 from AS/RS).

Element 5: 14 (Completion time of operation 1 of job 9.

Element 6: 2 (WC number j that performs the operation 1 of job 9. This is the current location of the AGV and job 9).

The appended four elements of the MPS are:

Element 1: 9 (Job number i).

Element 2: 2 (WC number j).

Element 3: 2 (Starting time of operation 1 of job 9 in WC 2, the operation starts after the elapse of time 2 or from time 3).

Element 4: 14 (Finishing time of operation 1 of job 9 in WC 2).

The elements of the AGVS appended are given below:

Element 1: 9 (The job that is loaded on the AGV).

Element 2: 0 (The starting location of AGV : AS/RS).

Element 3: 2 (The destination of AGV : WC2).

Element 4: 0 (AGV starts moving from AS/RS to WC2 on elapse of 0 time).

The above pices of information are obtained by manipulating the IS data file.

Step 4. Update the Status of WC, Job, AGVand AGV_task_list

The following variables change to new values. These changes update the status of WCs, jobs and AGV.

busy_sts(job 9):	14
busy_sts(WC2):	14
busy_sts(AGV):	2
loc_sts(AGV):	2 (i.e. WC2)
loc_sats(job 9):	2 (i.e. WC2)

The task vector of operation 2 of job 9 replaces the task vector of operation 1 of job 9 in WC 2 that is assigned for the AGV move and the production and the new *AGV_task_list* is obtained. The new *AGV_task_list* is given in Table 8.

Step 5. Termination

Steps 2 to 5 are repeated until all the tasks are exhausted from the task schedule matrix. This provides the complete IS, MPS and AGVS. The IS, MPS and AGVS obtained for SPT *vdr* is given in Tables 9, 10 and 11, respectively.

6. Performance Comparison

The proposed heuristic was tested on a set of 40 problems to compare the makespan performance of the *vdr*s. The problem size (i.e. the number of WCs 6 and number of jobs 10) of the data sets are the same. Table 12 furnishes the makespan time of the modified production schedules, which is adjusted for AGV operations or moves and resolved the conflicts with different *vdr*s. The results show that the STT rule provides the best solution (i.e. the best solution is the one that provides MPS with minimum shift from the original optimal solution) for 30 problems out of 40 problems processed. Also, the results

Table 11. AGV schedule (SPT vehicle dispatching rule).

Move	Element 1	Element 2	Element 3	Element 4	Move	Element 1	Element 2	Element 3	Element 4	Move	Element 1	Element 2	Element 3	Element 4
1	9	0	2	0	45	8	5	0	115	88	10	5	4	230
2	7	2	0	2	46	4	0	1	117	89	6	4	1	231
3	7	0	3	4	47	4	1	3	118	90	6	1	3	234
4	8	3	0	7	48	8	3	0	120	91	1	3	5	236
5	8	0	1	10	49	8	0	0	123	92	1	5	6	241
6	3	1	0	11	50	8	0	0	123	93	10	6	4	242
7	3	0	2	13	51	8	0	0	124	94	10	4	0	244
8	9	2	4	15	52	7	0	0	124	95	5	0	2	247
9	10	4	0	17	53	7	0	0	125	96	5	2	1	249
10	10	0	3	20	54	3	0	2	125	97	10	1	0	250
11	9	3	4	23	55	3	2	1	129	98	10	0	0	251
12	9	4	6	41	56	10	1	3	130	99	4	0	3	251
13	7	6	3	43	57	10	3	2	136	100	4	3	2	254
14	7	3	1	52	58	3	2	1	137	101	10	2	0	255
15	8	1	2	54	59	3	1	3	156	102	10	0	0	257
16	2	2	0	55	60	2	3	2	158	103	1	0	6	257
17	2	0	3	57	61	2	2	1	159	104	1	6	0	258
18	9	3	6	60	62	3	1	3	160	105	1	0	0	259
19	9	6	0	63	63	3	3	4	174	106	1	0	0	260
20	8	0	2	64	64	1	4	1	175	107	6	0	3	260
21	8	2	6	66	65	1	1	3	178	108	6	3	6	267
22	9	6	0	69	66	6	3	0	180	109	5	6	1	270
23	9	0	0	70	67	6	0	2	183	110	5	1	3	272
24	7	0	1	70	68	3	2	4	185	111	6	3	6	274
25	7	1	4	71	69	3	4	5	187	112	6	6	4	286
26	9	4	0	74	70	10	5	2	188	113	6	4	4	288
27	9	0	0	77	71	10	2	5	194	114	6	4	5	299
28	1	0	2	77	72	3	5	6	197	115	4	5	2	300
29	9	2	0	79	73	2	6	1	198	116	4	2	5	306
30	9	0	0	81	74	2	1	6	204	117	6	5	0	309
31	4	0	1	81	75	3	6	0	206	118	5	0	3	311
32	8	1	6	82	76	6	0	2	207	119	5	3	4	315
33	8	6	5	85	77	6	2	1	209	120	4	4	5	316
34	7	5	4	86	78	5	1	0	210	121	4	5	6	325
35	7	4	6	95	79	5	0	2	211	122	5	6	4	326
36	2	6	3	97	80	2	2	6	213	123	5	4	6	347
37	2	3	2	104	81	2	6	0	222	124	4	6	4	349
38	7	2	6	105	82	2	0	0	223	125	5	4	6	351
39	7	6	0	108	83	2	0	0	224	126	5	6	0	360
40	1	0	2	109	84	2	0	0	224	127	4	0	4	361
41	1	2	1	111	85	2	0	0	225	128	4	4	0	364
42	7	1	0	112	86	1	0	3	225	129	5	0	0	367
43	7	0	0	113	87	1	3	5	228	130	5	0	0	368
44	8	0	5	113										

of 6 of the remaining 10 cases are very close to the best solutions. This indicates that the STT rule outperforms the other *vdrs* significantly. It also observed that the utilisation of AGVs is more than 75%. The CPU time is about 0.4 s with a DELL 486 system. The above points indicate that the proposed methodology is applicable for FMS environments.

7. Conclusions

This paper attempts to link the AGV schedule with the production schedule. A heuristic algorithm is proposed to derive an optimal integrated schedule for makespan criterion, which gives a modified production schedule that integrates the AGV moves (AGV schedule) with minimum right shift, from the original production schedule. Since it is possible to get off-line schedules for both production and AGV, the control becomes easier. The proposed heuristic uses *vdrs* to resolve conflicts that arise during schedule generation. The proposed heuristic was tested on a set of 40 problems to compare the makespan performance of the *vdrs*. The test results indicate that the STT rule outperforms the other *vdrs* significantly. It is also observed that the utilisation of the AGV is more than 70% with the STT rule. The CPU time is also very reasonable.

In this paper, the AGV system is modelled with one AGV that operates in a single closed loop. This is a valid assumption provided the manufacturing system adopts a tandem configuration with many cells connected by individual and separate AGV loops. When the conventional AGV guide-path configuration is followed, the number of AGVs depends upon the frequency and volume of parts and cannot be limited to one. The heuristic is tested with four *vdrs* on a pre-specified layout.

Table 12. Performance comparison of different dispatching rules.

Example number	Makespan of schedules					Best result with rule
	Original production	Modified production schedule				
		SPT	LPT	FTT	STT	
1	299	368	385	418	341	STT
2	230	322	311	341	291	STT
3	185	259	315	267	243	STT
4	255	332	321	351	312	STT
5	295	384	341	333	365	FTT
6	212	314	274	304	263	STT
7	200	250	239	261	256	LPT
8	245	295	293	293	287	STT
9	405	451	417	439	425	LPT/STT
10	220	266	268	281	249	STT
11	191	319	283	312	244	STT
12	213	296	283	334	287	LPT/STT
13	196	273	290	342	215	STT
14	186	306	296	303	235	STT
15	192	301	325	304	255	STT
16	174	279	304	325	284	SPT/STT
17	253	275	310	297	284	SPT/STT
18	215	314	299	339	306	LPT/STT
19	270	316	314	314	257	STT
20	291	352	380	377	344	STT
21	309	373	383	370	381	FTT
22	265	331	296	293	287	STT
23	185	259	315	267	243	STT
24	270	354	304	366	337	LPT
25	295	385	339	371	339	STT
26	203	280	278	319	244	STT
27	185	261	247	260	230	STT
28	235	313	293	308	278	STT
29	395	438	432	445	425	STT
30	240	282	275	294	261	STT
31	196	304	294	335	252	STT
32	200	297	286	317	260	STT
33	191	278	276	286	230	STT
34	194	284	257	192	248	STT
35	192	308	288	296	249	STT
36	190	328	307	330	292	STT
37	249	371	327	351	325	STT
38	239	286	356	356	305	SPT/STT
39	245	331	352	391	268	STT
40	288	298	336	344	290	STT

The proposed heuristic can be extended and analysed for multi-AGV systems with dispatching rules such as LIV, NV, RV etc., with a few more *vdr*s, and with different layouts.

References

1. D. D. Bedworth and J. E. Bailey, Integrated Production Control Systems, 2nd edn, John Wiley, Singapore, pp. 77, 1987.
2. A. Eynan and M. J. Rosenblatt, "An interleaving policy in automated storage/retrieval systems", International Journal of Production Research, 31(1), pp. 1–18, 1993.
3. R. E. King, T. M. Hodgson and F. W. Chafee, "Robot task scheduling in a flexible manufacturing cell", IEE Transactions, 25(2), pp. 30–37, 1993.
4. J. A. Buzacott, "Modelling automated manufacturing systems", Proceedings of Fall Industrial Engineering Conference, pp. 130–136, 1983.
5. R. A. Pierson, "Adapting horizontal material handling systems to flexible manufacturing set-ups", Industrial Engineering, pp. 175–181, 1984.
6. Y. A. Bozer and M. M. Srinivasan, "Tandem configurations for AGV systems offer simplicity and flexibility", Industrial Engineering, pp. 23–27, 1989.
7. N. N. Krishnamoorthy, R. Batta and M. H. Karwan, "Developing conflict-free routes for automated guided vehicles", Operations Research Society of America, 41(6), pp. 1077–1090, 1993.
8. G. Uluso and U. Bilge, "Simultaneous scheduling of machines and automated guided vehicles", International Journal of Production Research, 31(12), pp. 2857–2873, 19932.
9. B. Giffler and G. L. Thompson, "Algorithms for solving production scheduling problems", International Journal of Operations Research, 8, pp. 487–503, 1960.
10. S. French, Sequencing and Scheduling, Ellis Horwood, London, 1982.
11. Y. A. Bozer and M. M. Srinivasan, "Tandem configurations for automated guided vehicle systems and the analysis of single vehicle loops", IIE Transactions, 23(1), pp. 72–82, 1991.
12. P. J. Egbelu and J. M. A. Tanchoco, "Potentials for bi-directional guided-path for AGV based systems", International Journal of Production Research, 24, pp. 1075–1097, 1986.
13. J. J. Spinelli, "The effects of load/unload times and networking zoning on an AGV system", Master's Thesis, Department of Industrial and Management Systems Engineering, Pennsylvania State University, 1987.
14. N. Raman, F. B. Talbolt and R. V. Rachamadgu, "Simultaneous scheduling of machines and material handling devices in automated manufacturing", Proceedings of the 2nd ORSA/TIMS Conference on FMS, Amsterdam, pp. 455–466, 1986.
15. M. M. Srinivasan, Y. A. Bozer and M. Cho, "Trip-based material handling systems: throughput capacity analysis", IIE Transactions, 26(1), pp. 70–89, 1994.
16. P. Mellor, "A review of job shop scheduling", Operations Research Quarterly, 17(2), pp. 161–171, 1966.