

A Deadlock-Avoidance Dispatching Method for Multiple-Load AGVs Based Transportation System

XIAO Haining¹, WU Xing^{2*}, ZOU Ting³, ZHAI Jingjing²

1. School of Mechanical Engineering, Yancheng Institute of Technology, Yancheng 224051, P. R. China;

2. College of Mechanical and Electrical Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, P. R. China;

3. Department of Mechanical Engineering, Memorial University of Newfoundland, St. John's A1B 3X5, Canada

(Received 29 September 2020; revised 23 December 2020; accepted 2 February 2021)

Abstract: In recent years, multiple-load automatic guided vehicle (AGV) is increasingly used in the logistics transportation fields, owing to the advantages of smaller fleet size and fewer occurrences of traffic congestion. However, one main challenge lies in the deadlock-avoidance for the dispatching process of a multiple-load AGV system. To prevent the system from falling into a deadlock, a strategy of keeping the number of jobs in the system (NJIS) at a low level is adopted in most existing literatures. It is noteworthy that a low-level NJIS will make the processing machine easier to be starved, thereby reducing the system efficiency unavoidably. The motivation of the paper is to develop a deadlock-avoidance dispatching method for a multiple-load AGV system operating at a high NJIS level. Firstly, the deadlock-avoidance dispatching method is devised by incorporating a deadlock-avoidance strategy into a dispatching procedure that contains four sub-problems. In this strategy, critical tasks are recognized according to the status of workstation buffers, and then temporarily forbidden to avoid potential deadlocks. Secondly, three multi-attribute dispatching rules are designed for system efficiency, where both the traveling distance and the buffer status are taken into account. Finally, a simulation system is developed to evaluate the performance of the proposed deadlock-avoidance strategy and dispatching rules at different NJIS levels. The experimental results demonstrate that our deadlock-avoidance dispatching method can improve the system efficiency at a high NJIS level and the adaptability to various system settings, while still avoiding potential deadlocks.

Key words: automatic guided vehicle (AGV) dispatching; deadlock avoidance; multiple-load AGV system; critical tasks; multi-attribute rules

CLC number: TP278

Document code: A

Article ID: 1005-1120(2021)01-0181-12

0 Introduction

Transportation system based on AGVs has been adopted as efficient, flexible and agile material handling systems in different environments, e. g., manufacturing plants, warehouses, distribution centers, and transshipment terminals. The design and control processes of an AGV system involve quite a few issues, i.e., guide path designing^[1], vehicle dispatching^[2], path/route planning^[3], and traffic management^[4].

The AGV system dispatching problem has been a hot area of academic research since the mid-1980s. Various dispatching methods have been developed. These dispatching methods can be divided into two types: offline and online. For the offline type, the task and vehicle data for the dispatching problem are known in advance. The entire dispatching scheme can be optimized in advance. For example, Umar et al.^[5] investigated the simultaneous scheduling of machines and dispatching AGVs in a flexible manufacturing system (FMS) environment.

*Corresponding author, E-mail address: wustar5353@nuaa.edu.cn.

How to cite this article: XIAO Haining, WU Xing, ZOU Ting, et al. A deadlock-avoidance dispatching method for multiple-load AGVs based transportation system[J]. Transactions of Nanjing University of Aeronautics and Astronautics, 2021, 38(1): 181-192.

<http://dx.doi.org/10.16356/j.1005-1120.2021.01.018>

A mixed-integer nonlinear programming model was formulated firstly, and a hybrid multi-objective genetic algorithm was then proposed to optimize the makespan, AGV travel time, and penalty cost. Kumar et al.^[6] addressed simultaneous scheduling of both machines and AGVs with alternative machines for the makespan minimizing objective. However, in practice, the manufacturing environments are usually stochastic. A small change in job arrival time or a failure of an AGV may destroy the whole dispatching plan. In this sense, online dispatching methods based on heuristic dispatching rules have more practical robustness. Maniya et al.^[7] proposed a multi-attribute dispatching method based on the AHP/MGRA technique. Xiao et al.^[8] proposed a real-time deadlock-free multi-attribute dispatching method (RMDM) using three attributes: empty traveling distance, remaining space status of both input and output buffers. Nevertheless, most of these methods are only appropriate for a single-load AGV system.

Recently, several studies have showed that the utilization of multiple-load AGVs has many advantages, such as less traffic congestion^[9], smaller fleet size^[10-11], and increased system throughput^[12]. Hence, the application of multiple-load AGVs is becoming increasingly popular in modern factories, which also encourages the theoretical research activities in academic communities.

Ho et al.^[13] proposed a dispatching procedure for a multiple-load AGVs. In this procedure, task determination rules^[14], delivery dispatching rules^[14], pickup dispatching rules^[13,15] and job selection rules^[15] were designed for four problems, which were evaluated via computer simulation. The simulation results indicated that job selection rules and pickup dispatching rules interact with each other^[15]. Then, a multiple-attribute method^[16] was proposed to solve the pickup dispatching problem and the job selection problem simultaneously. In the consideration of these studies, a reinforcement learning-based approach was used to select the best dispatching rule for a multiple-load carrier system in a general assembly line^[17-18]. Although these methods were sufficiently flexible and efficient, the system deadlock problem was not addressed yet.

In fact, dispatching rules should be properly designed to ensure that a system is deadlock-free. There are two types of deadlock in the AGVs: routing deadlock caused by the path conflict between several AGVs and dispatching deadlock caused by insufficient buffer capacity of the system. The former was usually resolved by means of a zone control traffic management method^[19-20], while the latter should be handled by properly designing the dispatching methods.

There are two kinds of approaches to handle the deadlock problem: (1) deadlock detection and elimination; (2) deadlock-avoidance. In accordance with some studies^[21-23], deadlock-avoidance is more efficient than the other solution. A resource-oriented colored Petri net was used to model a single-load AGVs^[21], on which a deadlock-avoidance strategy was developed based. However, some studies^[22-23] indicate that Petri nets also have disadvantages in terms of complexity and state space explosion. A remaining buffer capacity-based deadlock-avoidance policy was proposed by Guan and Dai^[23] to ensure a deadlock-free system and a high-efficiency dispatching performance simultaneously. Although the simulation results proved the effectiveness of the policy, it is only suitable for single-load AGV system.

To the authors' knowledge, the problem of deadlock-avoidance dispatching of multiple-load AGVs is seldom addressed in the pertinent literature. Several researchers^[13-16] have tried to reduce the possibility of dispatching deadlock by using a traditional strategy, which keeps the number of jobs in the system (NJIS) at a low level. However, a low level NJIS will make the processing machine easier to idle, so it unavoidably reduces the system efficiency.

The motivation behind this article lies in developing a deadlock-avoidance dispatching method for the multiple-load AGVs. Our main contributions are twofold. On the one hand, a deadlock-avoidance strategy based on the remaining capacity of the workstation is designed to keep the multiple-load AGVs deadlock-free at a higher NJIS level. It can then be incorporated into a multiple-load AGVs dispatching procedure with four dispatching problems.

On the other hand, most of the existing dispatching rules for the dispatching problems are single-attribute rules, which only reflect one aspect of the system. In order to describe multiple aspects of the system, some system states, e.g. the traveling distance and the buffer capacity, are selected to design several multi-attribute dispatching rules. Finally, several simulation experiments are carried out to evaluate the performance of the proposed deadlock-avoidance strategy and the multi-attribute dispatching rules.

1 Problem Statement

A manufacturing system is composed of a logistics transportation sub-system, a processing sub-system and a set of jobs that need to be processed. The processing sub-system consists of a series of workstations. The logistics transportation sub-system includes a guide path network and numerous multiple-

load AGVs. Each job is handled under a set of precedence constraints. Fig.1 gives the layout of a manufacturing system as an example. There are fifteen workstations, including an input workstation, an output workstation, and thirteen processing workstations. Each processing workstation has a processing machine, an input buffer and an output buffer with finite capacity. The input workstation has only an output buffer, and the output workstation has only an input buffer. All jobs enter the system via the input workstation. When a job is treated in the manufacturing system, it needs to be transported by AGVs and processed by machines. After a series of transport and processing tasks, the job leaves the manufacturing system from the output workstation. Each job has its own task sequences. Generally, the workstation of each processing task is predefined, and this paper focuses on the dispatching of multiple-load AGVs.

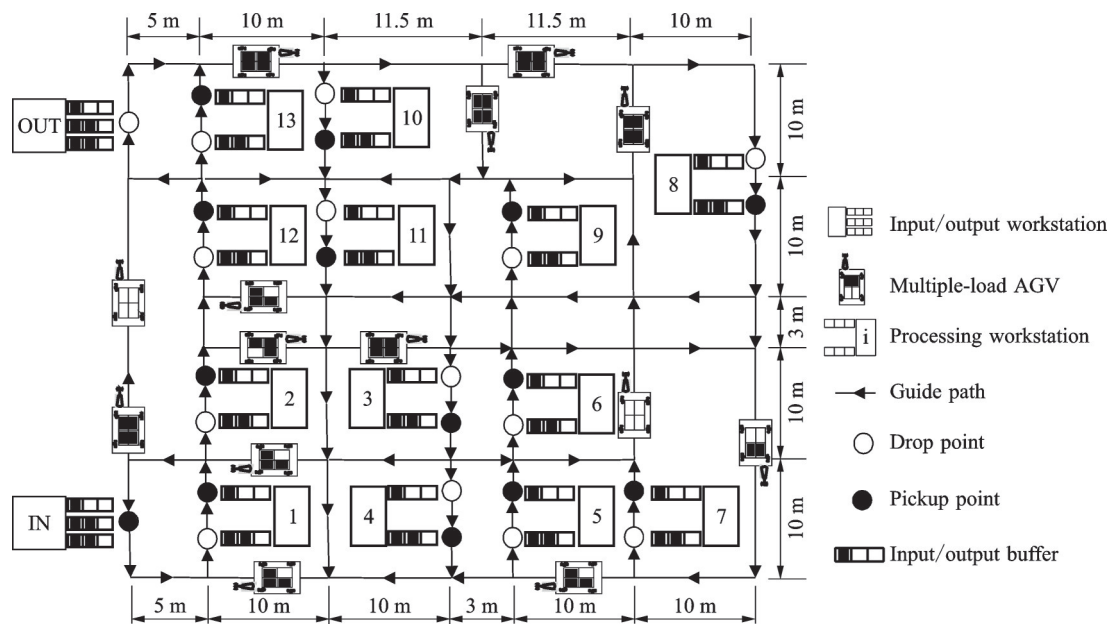


Fig.1 An example of manufacturing system

First, some important assumptions are given as follows.

(1) The processing machine of each workstation processes the jobs in the input buffer based on a first-come-first-served basis, while jobs in the output buffers can be picked up by the multiple-load AGVs in any order in accordance with scheduling requirements.

(2) A processing machine processes only one job at a time, and the conversion time for processing different types of job is neglected.

(3) All segments of the guide path network are unidirectional.

(4) All AGVs are multiple-load AGVs with the same capacity.

(5) All AGVs travel along the shortest path

between pick-up and drop points at a constant speed.

(6) The loading and unloading times for different types of jobs are the same. When one AGV is loading or unloading at a pick-up or drop point, other AGVs with the similar operations at this point must wait.

Before introducing the improved dispatching procedure, several notations and definitions are given as follows.

$W(m)$ —Workstation m .

$C_1(m)$ —Input buffer capacity of $W(m)$.

$c_1(m)$ —Current number of jobs in the input buffer of $W(m)$.

$C_o(m)$ —Output buffer capacity of $W(m)$.

$c_o(m)$ —Current number of jobs in the output buffer of $W(m)$.

$P(m)$ —Number of jobs on the processing machine of $W(m)$, with $P(m) = 0$ if the processing machine of $W(m)$ is empty; otherwise, $P(m) = 1$.

$L_o(m)$ —Number of jobs that have been assigned to a vehicle, and these jobs still in the output buffer in the output buffer of $W(m)$.

$L_1(m)$ —Number of jobs that have been assigned to a vehicle, and their destination workstation are $W(m)$.

$U(m)$ —Remaining capacity of $W(m)$. It can be calculated by

$$U(m) = C_1(m) + C_o(m) + 1 + L_o(m) - L_1(m) - c_1(m) - c_o(m) - P(m) \quad (1)$$

N_R —Number of multiple-load AGVs in the manufacturing system.

Blocked AGV—A multiple-load AGV is called blocked if it has picked up at least one job whose destination workstation has no remaining capacity.

N_B —Number of blocked multiple-load AGVs in the manufacturing system.

$N_{BM}(m)$ —Number of blocked multiple-load AGVs at the drop point of $W(m)$.

Active AGV—A multiple-load AGV that is not blocked.

TA—Task set waiting to be assigned to AGVs in the system, which can be denoted as

$$TA = \{T_{m_j} \cdots T_{ij} \cdots T_{j_m}\}, \text{ where } T_{ij} \text{ is a task}$$

with $W(j)$ as the destination workstation and $W(i)$ as the source workstation.

2 Deadlock-Avoidance Dispatching Procedure

Although Ho et al.^[15] proposed a dispatching procedure (DP) for multiple-load AGVs, they did not consider the dispatching deadlock caused by the finite buffer capacity of the workstation. To improve the DP, a deadlock-avoidance dispatching strategy is integrated. The complete flowchart of the improved dispatching procedure is shown in Fig.2.

2.1 Deadlock avoidance strategy

To activate a blocked AGV, other active AGVs need to release enough capacity by picking up some jobs in the output buffer of the destination workstation. In this sense, if $N_B = N_R$, no active AGV can release new capacity of the output buffer for other blocked AGVs. Consequently, the whole system enters a deadlock situation. To avoid the above-mentioned deadlock situations, it is necessary to ensure that there must be at least one active multiple-load AGV in the system. Thus, the deadlock avoidance strategy can be designed as follows.

(1) Only active multiple-load AGVs can receive new pick-up tasks. According to this rule, one AGV can take at most one job, whose destination workstation has no remaining capacity.

(2) If there is only one active multiple-load AGV in the system, some tasks should be temporarily forbidden. The temporarily forbidden task set is denoted as

$$TF = \{T_{ij} | N_{BM}(i) = 0 \text{ and } U(j) \leq 0 \text{ and } T_{ij} \in TA\} \quad (2)$$

Moreover, the available task set TC is given as

$$TC = TA/TF \quad (3)$$

Generally, multiple AGVs are used in a system. So, $N_R > 1$, and the deadlock avoidance property of the proposed strategy is proven as follows.

Proof 1

If the system has more than one active AGV, all tasks in the system belong to the available task set, and the system is deadlock-free.

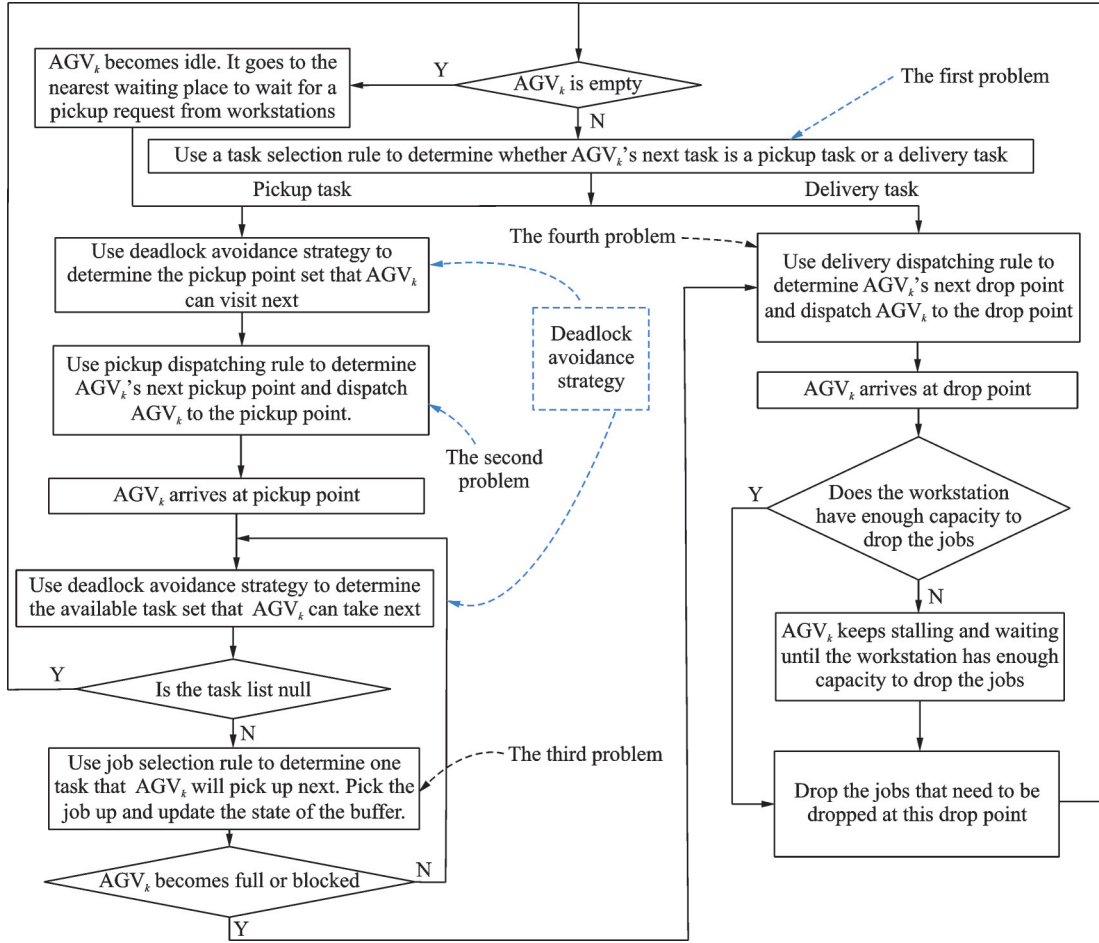


Fig.2 Flowchart of the deadlock-avoidance dispatching procedure

Proof 2

If the system has only one active multiple-load AGV, AGV_k . For any task, $T_{ij} \in TA$, according to the values of $U(j)$ and $N_{BM}(i)$, there are three possible cases:

Case 1 $U(j) > 0$ and $N_{BM}(i) \geq 0$

Case 2 $U(j) \leq 0$ and $N_{BM}(i) > 0$

Case 3 $U(j) \leq 0$ and $N_{BM}(i) = 0$

In case 1, the destination workstation of task T_{ij} has enough capacity. If task T_{ij} is assigned to AGV_k , then AGV_k will keep active.

In case 2, the destination workstation of task T_{ij} has no remaining capacity. AGV_k will be blocked if task T_{ij} is assigned to it. However, AGV_k will pick up T_{ij} on $W(i)$ and release a space for blocked AGVs in $W(i)$. As $N_{BM}(i) > 0$, the source workstation of task T_{ij} has blocked at least one multiple-load AGV, and one of the blocked AGVs, e.g., AGV_n , will become active. Since the system still

has at least one active AGV, it will not become deadlocked.

In case 3, the destination workstation of the task T_{ij} has no remaining capacity. If the task T_{ij} is assigned to AGV_k , then AGV_k will be blocked. Since $N_{BM}(i) = 0$, the source workstation has no blocked AGVs, and no blocked AGV can become active. Then, the system has no active AGV, and will enter the deadlock situation. However, if task T_{ij} is forbidden according to the deadlock avoidance strategy proposed above, it will not be assigned to AGV_k , and AGV_k will not be blocked.

Proof 3

As $N_R > 1$, if the system has only one active AGV, there must be at least one workstation has blocked AGVs. Then, all tasks in those workstations belong to case 2. All these tasks will not be forbidden. Thus, the situation in which all tasks are forbidden will not happen.

It is seen from Proofs 1, 2 and 3, the deadlock

avoidance strategy can keep at least one AGV active in the system and ensure that at least one task is not forbidden. Thus, the system is bound to deadlock-free.

2.2 Dispatching procedure

Based on the DP and the deadlock avoidance strategy proposed above, the basic flowchart of the improved dispatching procedure is shown in Fig. 2. The specific operations are described as follows.

Step 1 If one AGV (AGV_k) has just completed a pickup or delivery task, then considers the status of the AGV. If it is not empty, go to Step 2; otherwise, go to Step 3.

Step 2 Consider the first problem of the dispatching process—the task selection problem. Different task selection rules can be used to determine the next task for AGV_k . If the next task is a pickup task, go to Step 4; otherwise, go to Step 11.

Step 3 AGV_k becomes idle and waits for new pickup tasks. Once new pickup tasks arise, go to Step 4.

Step 4 The deadlock avoidance strategy is used to obtain the available task set $TC = \{T_{m_j} \cdots T_{ij} \cdots T_{j_m}\}$. Then, the available pickup points set, TPP, can be defined as

$$TPP = \{P_i | \forall T_{ij} \in TC\} \quad (4)$$

Step 5 Consider the second problem of the dispatching process—the pickup dispatching problem. The pickup dispatching rule is used to select a pickup point, P_i , from the available pickup point set TPP. Then, dispatch AGV_k to the pickup point P_i , and go to Step 6.

Step 6 Wait until AGV_k arrives at pickup point P_i , and then go to Step 7.

Step 7 The deadlock avoidance strategy is used to obtain the available task set $TC = \{T_{m_j} \cdots T_{ij} \cdots T_{j_m}\}$. Then, the available task set $TC(i)$ of workstation $W(i)$ can be defined as

$$TC(i) = \{T_{ij} | \forall T_{ij} \in TC\} \quad (5)$$

Step 8 If the available task set, $TC(i)$, is null, go to Step 1; otherwise, go to Step 9.

Step 9 Consider the third problem of the dispatching process—the job selection problem. The

job selection rule is used to select one task, T_{ij} , from the available task set. Then, $L_o(i)$ and $L_1(j)$ are updated accordingly.

Step 10 Check the status of AGV_k . If it is full or blocked, go to Step 11; otherwise, go to Step 7.

Step 11 Consider the fourth problem of the dispatching process—the delivery dispatching problem. The delivery dispatching rule is used to determine the next drop point D_j .

Step 12 Wait until AGV_k arrives at drop point D_j .

Step 13 Check the remaining capacity of workstation $W(j)$. If $W(j)$ does not have enough capacity to hold the job, then go to Step 14; otherwise, go to Step 15.

Step 14 AGV_k remains stationary until $W(j)$ has enough capacity to accept the job. Then, go to Step 15.

Step 15 AGV_k drops the jobs whose destination is $W(j)$. And AGV_k becomes active. Then, go to Step 1.

2.3 Rules adopted for the four problems

Ho et al.^[13-15] proposed some relevant rules for the four dispatching problems. These rules can also be adopted by the improved DP. However, most of these rules are single-attribute rules. Since a single attribute cannot reflect all aspects of the system state, we propose the following multi-attribute rules.

2.3.1 The multi-attribute rule for the pickup dispatching problem (MARP)

To reflect various aspects of the system state, the output buffer status and the travelling distance is chosen to calculate the pickup-dispatching utility values. Suppose the available task set is $TC = \{T_{m_j} \cdots T_{ij} \cdots T_{j_m}\}$ and the available pickup point set is $TPP = \{P_i | \forall T_{ij} \in TC\}$. The utility values between AGV_k and a pickup point P_i can be calculated by

$$F(P_i) = F_D(P_i) + F_O(P_i) \quad (6)$$

where $F_D(P_i)$ and $F_O(P_i)$ are the values of the two attributes. AGV_k will visit the pickup point with the maximum utility value.

$F_D(P_i)$ is the travelling distance attribute between AGV_k and pickup point P_i , and it can be obtained by

$$F_D(P_i) = \frac{MDP - D(P_i)}{MDP} \quad (7)$$

where $MDP = \text{MAX}(AGV_k, TPP)$ is the maximum distance between AGV_k and any pickup point in the available pickup point set TPP. $D(P_i)$ is the distance between AGV_k and pickup point P_i .

The output buffer status attribute $F_O(P_i)$ can be defined as

$$F_O(P_i) = \frac{c_o(i) - L_o(i)}{C_o(i)} \quad (8)$$

2.3.2 The multi-attribute rule for the job selection problem (MARJ)

Suppose AGV_k has arrived at pickup point P_i . The travelling distance and the input buffer status are chosen as the dispatching attributes of job selection. The available task set of workstations $W(i)$ is $TC(i) = \{T_{ij} | \forall T_{ij} \in TC\}$. The utility values between AGV_k and the job delivering task T_{ij} can be calculated by

$$F(T_{ij}) = F_D(T_{ij}) + F_I(T_{ij}) \quad (9)$$

where $F_D(T_{ij})$ and $F_I(T_{ij})$ are the values of the two attributes. AGV_k will pick up the job with the maximum utility value.

$F_D(T_{ij})$ is the travelling distance attribute between the destination of task T_{ij} and the destination of any job currently on AGV_k, and it can be calculated by

$$F_D(T_{ij}) = \frac{MDT - D(T_{ij})}{MDT} \quad (10)$$

where MDT is the maximum distance between any pair of drop points in the system. $D(T_{ij})$ is the minimum distance between the destination of task T_{ij} and that of any job currently on AGV_k. The input buffer status attribute $F_I(T_{ij})$ is defined as

$$F_I(T_{ij}) = \begin{cases} 1 - \frac{c_1(j) - L_1(j)}{C_1(j)} & C_1(j) > c_1(j) + L_1(j) \\ 0 & \text{else} \end{cases} \quad (11)$$

2.3.3 The multi-attribute rule for the delivery dispatching problem (MARD)

If AGV_k has picked up several jobs and its next task is a delivery task, the multi-attribute rule is used to identify which drop point AGV_k should visit first. Suppose the job set that AGV_k has picked up is $TJ = \{T_{mj} \cdots T_{ij} \cdots T_{jm}\}$. Then, the drop point set can be defined as $TPD = \{D_j | \forall T_{ij} \in TJ\}$. Travelling distance and input buffer status are selected to calculate the utility value between AGV_k and a drop point in TPD. The utility value is defined as

$$F(D_i) = F_D(D_i) + F_I(D_i) \quad (12)$$

where $F_D(D_j)$ and $F_I(D_j)$ are the values of the two attributes. AGV_k will visit the drop point with the maximum utility value first.

$F_I(D_j)$ is the travelling distance attribute between the drop point D_j and the current position of AGV_k, which can be calculated by

$$F_D(D_j) = \frac{MDD - D(D_j)}{MDD} \quad (13)$$

where $MDD = \text{MAX}(AGV_k, TPD)$ is the maximum distance between AGV_k and all drop points in the drop point set TPD. $D(D_j)$ is the distance between AGV_k and the drop point D_j .

The input buffer status attribute $F_I(D_j)$ is defined as

$$F_I(D_j) = \begin{cases} 1 - \frac{c_1(j) - L_1(j)}{C_1(j)} & C_1(j) > c_1(j) + L_1(j) \\ 0 & \text{else} \end{cases} \quad (14)$$

3 Simulation Experiments

3.1 Simulation model

To verify the performance of the deadlock avoidance strategy and the multi-attribute rules proposed in this paper, simulation tests are carried out to compare the TS with our strategy under eight combinations of rules. The rules adopted in our simulation are introduced as follows.

3.1.1 Rules adopted for the task selection problem

Ho et al.^[14] proposed three single-attribute

rules for this problem and compared their performance results. Simulation results showed that the DTF rules had the best throughput performance. Hence, the task selection rule adopted in our simulation experiments is DTF. According to this rule, if the AGV is partially loaded, it will always perform a delivery task.

3.1.2 Rules adopted for the pickup dispatching problem

Two rules are adopted in our simulation experiments. One is the proposed MARP, and the other is the greatest queue length (GQL) rule. According to Ref. [13], GQL is the best among the twelve pickup dispatching rules proposed in their paper. Based on the GQL rule, one AGV (AGV_k) will visit the pickup point that has the greatest number of jobs waiting at its output buffer.

3.1.3 Rules adopted for the job selection problem

The first rule adopted in our simulation experiments is MARJ. Additionally, Ho and Liu^[15] studied six job selection rules to evaluate their performances. Their work has showed that the identical destination first (IDF) rule had the best performance. Hence, the IDF rule is adopted as the second rule. When the IDF rule is used, one AGV (AGV_k) will pick up the job whose destination is the same as the next destination of any load current-

ly on AGV_k .

3.1.4 Rules adopted for the delivery dispatching problem

The first rule we adopted is MARD. The second is the shortest distance (SD) rule. Ho and Chien^[14] have proved that the SD rule is better than other delivery dispatching rules in all performance measures. This rule ensures that the job with the smallest distance between its destination and the current position of AGV will be delivered first.

The eight hybrid-rule sets are described in Table 1. An example of a manufacturing system is shown in Fig.1. The capacity of the buffer in the input workstation is 4, and that of the input and output buffers in every processing workstation is 2. The processing information of the job set is listed in Table 2. The processing time for every job in every workstation is 90 s.

Table 1 All hybrid-rule sets adopted in this study

Combination code	Rules adopted for each problem
1111	DTF+MARP+MARJ+MARD
1112	DTF+MARP+MARJ+SD
1121	DTF+MARP+IDF+MARD
1122	DTF+MARP+IDF+SD
1211	DTF+GQL+MARJ+MARD
1212	DTF+GQL+MARJ+SD
1221	DTF+GQL+IDF+MARD
1222	DTF+GQL+IDF+SD

Table 2 The processing information of the job set

Job type	Operation sequence															Mix ratio/ %
	1	2	12	8	4	14	13	10	7	6	3	5	9	11	15	10
A	1	2	12	8	4	14	13	10	7	6	3	5	9	11	15	10
B	1	3	13	9	5	4	14	11	8	7	2	6	10	12	15	10
C	1	4	14	10	6	5	2	12	9	8	3	7	11	13	15	10
D	1	5	2	11	7	6	3	13	10	9	4	8	12	14	15	10
E	1	6	2	13	8	7	4	14	11	10	3	5	9	12	15	10
F	1	7	3	13	9	8	5	2	12	11	4	6	10	14	15	10
G	1	8	4	14	10	9	6	3	13	12	2	5	11	7	15	10
H	1	9	5	3	11	10	7	4	14	13	2	6	8	12	15	10
I	1	10	6	2	12	11	8	5	4	14	3	7	9	13	15	10
J	1	11	7	3	13	12	9	6	5	2	4	8	10	14	15	10

There are ten multiple-load AGVs in the system. Each AGV has a 4-item loading capacity and

travels at the speed of 1 m/s. It takes 30 s for one AGV to perform a loading or unloading operation.

The software Tecnomatix Plant Simulation 15.0 is utilized in our simulation tests, with a software interface as shown in Fig.3.



Fig.3 Simulation experiment interface

3.2 Tests using the traditional strategy

First, the traditional strategy is used in the simulation tests. The simulation duration is 96 h, and each kind of test repeats 10 times. The deadlock times for the traditional strategy with different hybrid-rule sets are shown in Fig.4. and the average throughput for this strategy is shown in Fig.5.

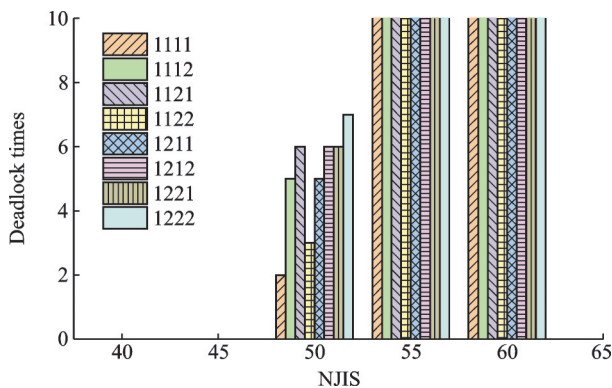


Fig.4 Deadlock times for the traditional strategy under all hybrid-rule sets

Moreover, the simulation results show that the hybrid-rule set with code 1222 enters deadlock most probably for all NJIS levels. When the NJIS level is 55, this hybrid-rule set enters deadlock very early and produces no jobs. Moreover, the average throughput of the hybrid-rule set with code 1111 is 0.8%—64.4% higher than the other hybrid-rule sets. Based on the obtained results, it is obvious that the multi-attribute rules can synthesize the mul-

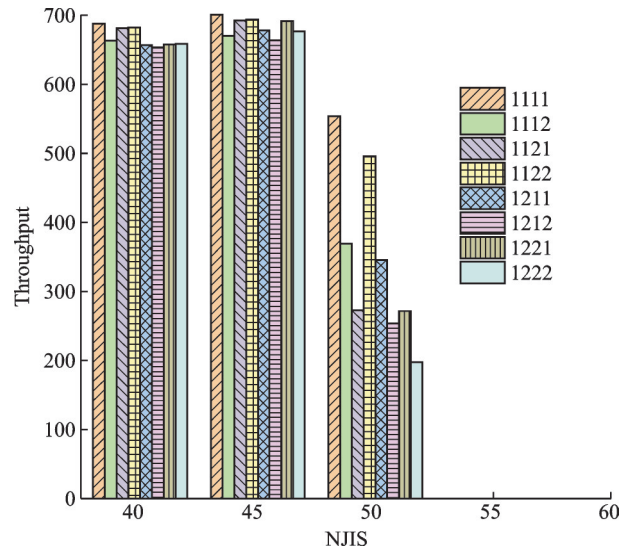


Fig.5 Average throughput for the traditional strategy under all hybrid-rule sets

iple aspects of the system.

3.3 Tests using the deadlock avoidance strategy

In this subsection, simulation tests are conducted to validate the deadlock-avoidance strategy proposed in this paper. The system parameter settings are the same as the previous one. When the deadlock-avoidance strategy is used, no deadlock occurs for any hybrid-rule set at any NJIS level. The average throughput for the deadlock-avoidance strategy with different hybrid-rule sets is shown in Fig.6.

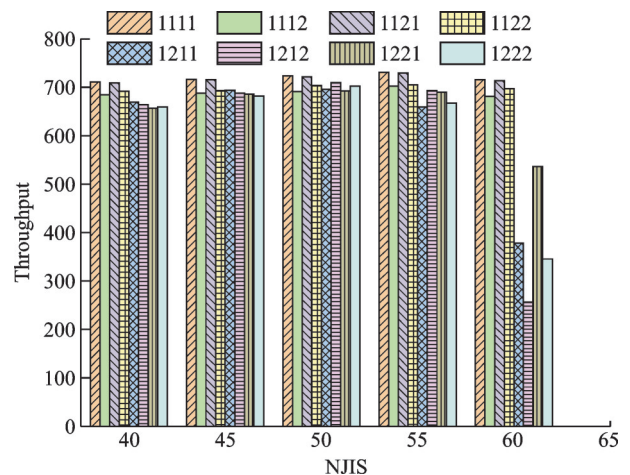


Fig.6 Average throughput for the deadlock avoidance strategy under all hybrid-rule sets

In Fig.6, when the deadlock avoidance strategy is used, the best NJIS levels for all hybrid-rule sets are more than 50. In contrast, if the traditional

strategy is used, all hybrid-rule sets cannot prevent a deadlock. It is obvious that the deadlock-avoidance strategy is indispensable. The comparison of Figs.5 and 6 shows that, when the deadlock-avoidance strategy is adopted, the average throughput for all hybrid-rule sets at each NJIS level increase. Take the hybrid-rule set with code 1121 for example, when using the proposed deadlock-avoidance strategy, the optimal average throughput is 729.4, which is 5.2% higher than 693.1 with the traditional strategy. This result implies that the proposed deadlock-avoidance strategy can make full use of system resources and improve the performance of dispatching rules.

As shown in Fig.6, the hybrid-rule set with code 1111 performs better than any other set at each NJIS level. The optimal average throughput of the hybrid-rule set with code 1111 is 730.5, which is 0.2%—9.7% higher than the other hybrid-rule sets. This result indicates that the multi-attribute rules are adaptive to various system settings and more efficient than single-attribute rules for multiple-load AGVs with finite buffer capacity constraints.

The simulation results also show that the NJIS level has a significant effect on the system throughput. Take the hybrid-rule set with code 1221 for example, the system throughput, working rate, blocking rate and starvation rate of the processing machines are shown in Fig.7. The throughput performance is directly proportional to the average work-

ing rate of the processing machines. If the NJIS level is too low, the processing machines are frequently starved. On the other hand, if the NJIS level is too high, the processing machines are likely to become blocked as the output buffer is frequently full. Both of these two situations will reduce the average working rate of the processing machines; thus, the efficiency of the processing sub-system being decreased. It is obvious that the NJIS level has a significant effect on the system throughput. Every hybrid-rule set has a best NJIS level. When operated below or above this level, the system efficiency will be reduced.

4 Conclusions

The dispatching problem of multiple-load AGVs with finite buffer capacity constraints is presented. The deadlock-avoidance dispatching strategy based on the remaining capacity of the workstation is integrated with a multiple-load AGV dispatching procedure. The following findings can be obtained from the simulation results.

A proper NJIS level is critical to manufacturing systems with finite buffer capacity constraints. Although a lower NJIS level can prevent the system from reaching deadlock, this approach will also reduce the system efficiency. The dispatching problem of multiple-load AGVs with finite buffer capacity constraints is addressed for a manufacturing system operated at a high NJIS level. In order to improve the system efficiency at a higher NJIS level while still avoiding deadlock, the deadlock-avoidance dispatching strategy based on the remaining capacity of the workstation is integrated with a multiple-load AGV dispatching procedure. Simulation results show that the proposed deadlock-avoidance strategy can make full use of system resources and improve the performance of dispatching rules. Moreover, the simulation results also show that the optimal average throughput of the hybrid-rule set with code 1111 is 0.2%—9.7% higher than the other hybrid-rule sets. This indicates that the proposed multi-attribute rules are adaptive to various system settings and

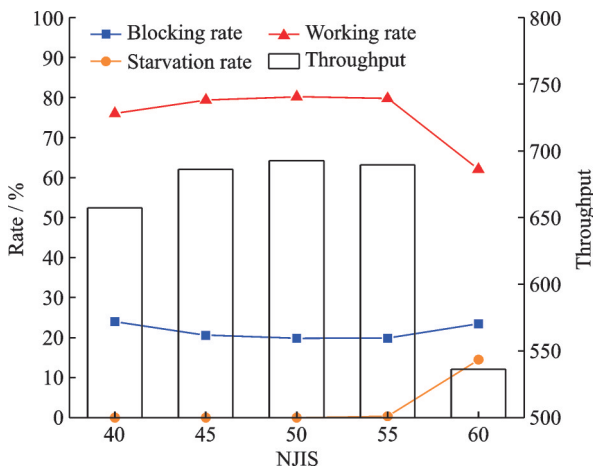


Fig.7 Average system throughput, working rate, blocking rate and starvation rate of the processing machines under the hybrid-rule set with code 1221

more efficient than single-attribute rules for multiple-load AGVs with finite buffer capacity constraints. It is hopeful that these research findings can be helpful for solving multiple-load AGV dispatching problems in similar industrial scenarios. However, the intrinsic relationship of the NJIS level with the buffer size of the workstations or the multiple-load AGVs is still not fully clarified yet. Hence, it will be investigated further in our future research work.

References

- [1] XIAO H N, WU X, ZENG Y, et al. A CEGA-based optimization approach for integrated designing of a unidirectional guide-path network and scheduling of AGVs[J]. *Mathematical Problems in Engineering*, 2020, 2020(2): 1-16.
- [2] DAI Min, ZHANG Yuwei, ZENG Li. Integrated scheduling of machines and AGVs in green job shop [J]. *Journal of Nanjing University of Aeronautics & Astronautics*, 2020, 52(3): 468-477. (in Chinese)
- [3] YANG L, WANG P, ZHANG Y. Coordinated path planning for UAVs based on sheep optimization[J]. *Transactions of Nanjing University of Aeronautics and Astronautics*, 2020, 37(5): 816-830.
- [4] XIAO H N, WU X, QIN D J, et al. A collision and deadlock prevention method with traffic sequence optimization strategy for UGN-based AGVs[J]. *IEEE Access*, 2020, 8: 209452-209470.
- [5] UMAR U A, ARIFFIN M K A, ISMAIL N, et al. Hybrid multiobjective genetic algorithms for integrated dynamic scheduling and routing of jobs and automated-guided vehicle (AGV) in flexible manufacturing systems (FMS) environment[J]. *The International Journal of Advanced Manufacturing Technology*, 2015, 81(9/10/11/12): 2123-2141.
- [6] KUMAR M V S, JANARDHANA R, RAO C S P. Simultaneous scheduling of machines and vehicles in an FMS environment with alternative routing[J]. *The International Journal of Advanced Manufacturing Technology*, 2011, 53(1/2/3/4): 339-351.
- [7] MANIYA K D, BHATT M G. A multi-attribute selection of automated guided vehicle using the AHP/MGRA technique[J]. *International Journal of Production Research*, 2011, 49(20): 6107-6124.
- [8] XIAO Haining, LOU Peihuang, MAN Zengguang, et al. Real-time multi-attribute dispatching method for automatic guided vehicle system[J]. *Computer Integrated Manufacturing Systems*, 2012, 18(10): 2224-2230. (in Chinese)
- [9] DU L, KE S, WANG Z, et al. Research on multi-load AGV path planning of weaving workshop based on time priority[J]. *Mathematical Biosciences and Engineering: MBE*, 2019, 16(4): 2277-2292.
- [10] CHAWLA V K, CHANDA A K, ANGRA S. Multi-load AGVs scheduling by application of modified memetic particle swarm optimization algorithm[J]. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, 2018, 40(9): 436.
- [11] CHAWLA V K, CHANDA A K, ANGRA S. Scheduling of multi load AGVs in FMS by modified memetic particle swarm optimization algorithm[J]. *Journal of Project Management*, 2018, 3(1): 39-54.
- [12] ANGRA S, CHANDA A, CHAWLA V. Comparison and evaluation of job selection dispatching rules for integrated scheduling of multi-load automatic guided vehicles serving in variable sized flexible manufacturing system layouts: A simulation study[J]. *Management Science Letters*, 2018, 8(4): 187-200.
- [13] HO Y C, LIU H C. A simulation study on the performance of pickup-dispatching rules for multiple-load AGVs[J]. *Computers & Industrial Engineering*, 2006, 51(3): 445-463.
- [14] HO Y C, CHIEN S H. A simulation study on the performance of task-determination rules and delivery-dispatching rules for multiple-load AGVs[J]. *International Journal of Production Research* 2006, 44(20): 4193-4222.
- [15] HO Y C, LIU H C. The performance of load-selection rules and pickup-dispatching rules for multiple-load AGVs[J]. *Journal of Manufacturing Systems*, 2009, 28(1): 1-10.
- [16] HO Y C, LIU H C, YIH Y. A multiple-attribute method for concurrently solving the pickup-dispatching problem and the load-selection problem of multiple-load AGVs[J]. *Journal of Manufacturing Systems*, 2012, 31(3): 288-300.
- [17] CHEN C, XIA B, ZHOU B, et al. A reinforcement learning based approach for a multiple-load carrier scheduling problem[J]. *Journal of Intelligent Manufacturing*, 2015, 26(6): 1233-1245.
- [18] ZHOU B, XU J. An adaptive SVM-based real-time scheduling mechanism and simulation for multiple-load carriers in automobile assembly lines[J]. *International Journal of Modeling, Simulation, and Scientific Computing*, 2017, 8(4): 1750048.
- [19] REVELIOTIS S. An MPC scheme for traffic coordination in open and irreversible, zone-controlled, guide-path-based transport systems[J]. *IEEE Transactions*

on Automation Science and Engineering, 2020, 17(3): 1528-1542.

- [20] LI Q, POGROMSKY A, ADRIAANSEN T, et al. A control of collision and deadlock avoidance for automated guided vehicles with a fault-tolerance capability[J]. International Journal of Advanced Robotic Systems, 2016, 13(2): 1-24.
- [21] WU N, ZHOU M C. Modeling and deadlock avoidance of automated manufacturing systems with multiple automated guided vehicles[J]. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 2005, 35(6): 1193-1202.
- [22] LI Z W, WU N Q, ZHOU M C. Deadlock control of automated manufacturing systems based on Petri nets—A literature review[J]. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 2012, 42(4): 437-462.
- [23] GUAN X P, DAI X Z. Deadlock-free multi-attribute dispatching method for AGV systems[J]. The International Journal of Advanced Manufacturing Technology, 2009, 45(5): 603-615.

Acknowledgements This work was supported by the National Natural Science Foundation of China (Nos. 52005427, 61973154), the National Defense Basic Scientific Research Program of China (No. JCKY2018605C004), the Natural Science Research Project of Jiangsu Higher Education Institutions (Nos. 19KJB510013, 18KJA460009), and the Foundation of Graduate Innovation Center in Nan-

jing University of Aeronautics and Astronautics (No. KFJJ20190516).

Authors Dr. XIAO Haining received his B.S degree in Mechanical Engineering from Yancheng Institute of Technology in 2007 and Ph.D. degree in Mechanical and Electronic Engineering from Nanjing University of Aeronautics and Astronautics (NUAA), Nanjing, China, in 2014. He is currently a lecturer in Yancheng Institute of Technology. His research interests include coordinated control and intelligent scheduling of multi-AGV systems.

Dr. WU Xing received his B.S. and Ph.D. degrees in Mechanical Engineering from NUAA in 2004 and 2010, respectively. In 2010, he joined the College of Mechanical and Electrical Engineering, NUAA, where he is currently an associate professor. He was a visiting professor at the Centre for Intelligent Machines, McGill University, Canada, from November 2016 to November 2017. His research interests include design, guidance, navigation and control of mobile robots, and coordinated control and intelligent scheduling of multi-robot systems.

Author contributions Dr. XIAO Haining and Dr. WU Xing conceived the idea of this paper and wrote the manuscript. Dr. ZOU Ting contributed to the conclusions for the study. Ms. ZHAI Jingjing conducted the simulation model and experiments. All authors commented on the manuscript draft and approved the submission.

Competing interests The authors declare no competing interests.

(Production Editor: WANG Jing)

基于多载量AGV的运输系统死锁避免调度方法

肖海宁¹, 武星², 邹婷³, 翟晶晶²

(1. 盐城工学院机械工程学院, 盐城 224051, 中国; 2. 南京航空航天大学机电学院, 南京 210016, 中国;

3. 纽芬兰纪念大学机械工程系, 圣约翰市 A1B 3X5, 加拿大)

摘要:近年来,多载量自动导引车(Automatic guided vehicle, AGV)系统因其交通拥堵率低、车队规模性小等优势,在物流运输领域的应用愈加广泛。实现死锁避免一直是多载量AGV系统调度过程中面临的主要挑战之一。为了避免系统死锁,现有文献普遍采用了低在制品数策略。然而,该策略会使加工机床易于待料停产,最终降低系统效率。为此,提出了一种适用于高在制品数AGV系统的死锁避免方法。首先,在包含4个子问题的调度过程中融入了一个死锁避免策略,该策略根据系统缓存区状态动态识别并临时禁止关键任务运行以避免潜在死锁。其次,为了提高系统的效率,综合考虑输入/输出缓存区状态、行驶距离等属性设计了3条多属性调度规则。最后,开发了一个仿真系统来评估所提出的死锁避免策略和多属性调度规则在不同在制品数上的性能。仿真结果表明,所提出的死锁避免调度方法不仅可以避免潜在的死锁,而且可以在较高的在制品数量上提高系统的效率,具有较高的系统适应性。

关键字:自动导引车调度;死锁避免;多载量自动导引车系统;关键任务;多属性调度规则