ISSN: 2414-1895 DOI: 10.6919/ICJE.202010_6(10).0034

AGV Real-time Scheduling Based on Multi-feature and Tree-Convolutional Neural Network

Renjie Bo[1,a], Jin Zhu[1]

[1]Institute of Logistics Science & Engineering, Shanghai Maritime University, Shanghai 201306, China.

[a]372050269@qq.com

Abstract

Aiming at the dynamic scheduling problem of automated terminal AGV, a multi-feature-based dispatching strategy is designed. Through advanced simulation evaluation of the multiple features of each container operation, the preference function is used to find the best candidate job. Then the best job is paired with each of the remaining jobs to compose the training samples, and the tree convolutional neural network is used to learn the training samples incrementally and update the dispatching strategy for the next scheduling process. The scheduling results of 12-30 AGVs under Tree-CNN, .CNN and BP. When the number of AGV increases to 24, the tree-CNN solution is used to obtain the best QC average operation time, and the empty distance is reduced by 10.3% to reach the optimum. As the number of AGV increases and the workload increases, the advantages of using Tree-CNN to solve over CNN and BP will become more and more obvious. Finally, it is concluded that the multi-feature dispatching strategy based on Tree-CNN is suitable for whose number of AGV between 24 and 30 by analyzing the impact of weights on performance.

Keywords

Dispatching Strategy, Automated Guided Vehicle, Tree Convolutional Neural Network, Real-time Scheduling.

1. Introduction

AGV has the characteristics of autonomous positioning, autonomous navigation, information interaction, etc., which can effectively reduce the turnaround time and operating cost of containers, combined with the optimized scheduling of AGV real-time data information[1][2].

At present, the scheduling research for AGV generally uses heuristic algorithms to solve the problem under the premise of known container operation task information[3][4], but this traditional optimization method cannot meet the dynamic changes of automated terminals. Therefore, in recent years, the real-time dynamic scheduling method based on machine learning has received more and more attention, and the optimal strategy of each state is learned through real data or simulation[5]. At present, neural networks are widely used in classification, clustering, and data processing[6]. Xie Mingli et al.[7] proposed a neural network weight optimization algorithm based on maximum entropy, which solved the problem that traditional neural networks tend to fall into local optimality when performing scheduling strategies. Akyol et al.[8] proposed a decision-making system, based on neural network, which acquires data through simulation experiments and trains the neural network so that the neural network can select the best decision based on the given parameters. Choe et al.[9][10] proposed an online learning assignment strategy, and aimed at the average delay time of quay cranes and the shortest AGV empty distance, using back propagation to train the preference strength between

ISSN: 2414-1895 DOI: 10.6919/ICJE.202010_6(10).0034

two containers and realize real-time assignment. However, there are too many weights and too much heavy calculation in the BP neural network, requiring a large quantity of samples for training, which will affect the solution rate. Xin et al.[11] aimed to minimize the total completion time of loading and unloading containers, and proposed a method of equipment rescheduling based on the current situation. Ren et al.[12] adopted reinforcement learning algorithms to establish Q learning and off-line learning modules for multiple scheduling strategies. After each scheduling, the next state scheduling strategy was dynamically given. However, reinforcement learning using nonlinear functions often results in divergence due to different features, causing poor convergence. In consequence, convolutional neural networks can be used to extract features from training data and obtain a nonlinear real-time model through incremental learning of the decision tree.The use of neural network simulation fusion technology not only solves the linear inseparability problem, but also improves the generalization ability of the decision tree.

In conclusion, aiming at the "Catastrophic Oblivion" problem in automated terminal AGV scheduling and online learning, once a new data set is used to train an existing model, the model will lose its ability to recognize the original data set. Considering the constraints of the AGV's driving speed and the urgency of each container, establish an AGV real-time scheduling model with the shortest average operation time of quay crane (QC) and the shortest AGV empty distance, using tree convolutional neural network in online learning of multi-feature assignment strategies. A comparative experiment, based on the multi-feature assignment strategy is designed to compare the average QC operating time and AGV empty distance under the tree convolutional neural network, convolutional neural network and back propagation. And analyzed the influence of weight change on the performance of the dispatching strategy.

2. Multi-feature dispatching model

2.1 Model framework and Variable setting

let define a container, when it needs QC for loading and unloading without any AGV dispatched, then this container is a candidate operation, and the following assumptions are made:

A QC corresponds to multiple container yards, and the quay cranes and yard cranes served by the AGV are not fixed;

AGV only transports one container at a time;

All container assignment plans are known;

When the AGV is out of work, it will return to the queue area;

When the AGV completes the current job, it triggers the scheduling. The first step is to collect candidate jobs from the loading or unloading sequence of each QC. The online learning algorithm uses the current scheduling strategy to select a job from the candidate jobs, and then assigns the job to the AGV requesting the new job. The current scheduling strategy the job selected from the candidate jobs is not necessarily the best one. In particular, the job starts with a random initial strategy. In the second step, based on the multiple characteristics of each candidate job, each candidate job is evaluated by short-term advanced simulation of continuous job assignments(L), and the preference function is used to check the evaluation values of all candidate jobs to determine the best one. The best job is paired with the remaining jobs one by one to form training samples. Finally, the Tree-CNN is used to learn the training samples online and update the assignment strategy in this simulation, and apply the assignment strategy to the next scheduling process. In order to illustrate the model, the following variables are quoted. The flowchart is shown in Figure 1.

J is candidate job set.

π is dispatching strategy.

T is sampling pool.

R is sample limit.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202010_6(10).0034

L is the amount of simulation jobs.

I is a container job.

J^* is the best job.

C is newly generated job training sample.

V is the evaluation array of candidate job set.

ω is weight.

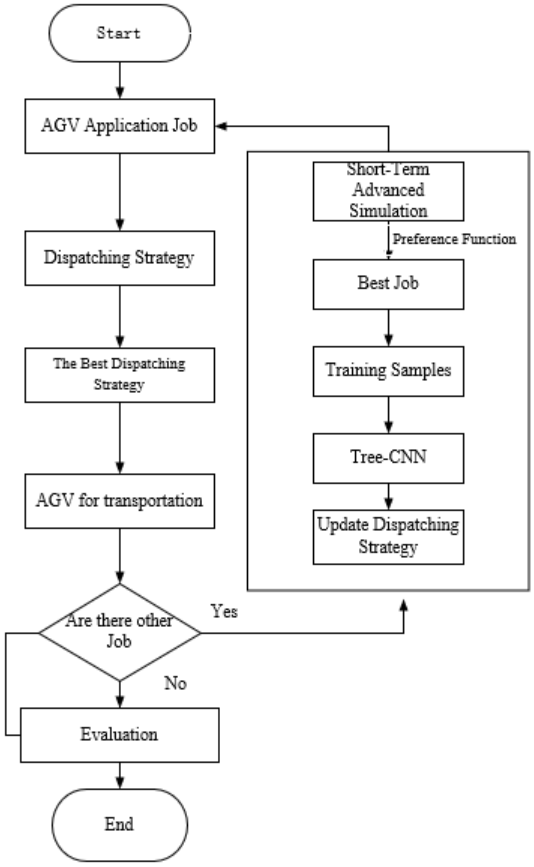

Fig.1 Dispatching flowchart of AGV real-time

2.2 Multiple-Features Dispatching Strategy

When assigning jobs to AGVs, the best job is selected by evaluating various criteria for candidate jobs, some of which are designed to minimize the empty distance of the AGV and minimize the average operation time of QC. This paper uses 7 status to represent a candidate operation for loading and unloading ships, as shown in Table 1. Each standard will be described in detail as below.

Table 1 Dispatching criteria that promote productivity enhancement and reduction of empty travel

Feature Criteria	Description
$C_1(x)$	Urgency of the target job
$C_2(x)$	Differenc between the arrival times of the current AGV and the other competing AGVs
$C_3(x)$	Empty travel distance
$C_4(x)$	Negative of loaded travel distance
$C_5(x)$	Indication of loading or discharging
$C_6(x)$	Negative of average QC delay
$C_7(x)$	Remaining workload of the target block relative to the average remaining workload of all the blocks

ISSN: 2414-1895 DOI: 10.6919/ICJE.202010_6(10).0034

Let X be the container of the candidate job, and then build a d-dimensional vector about the candidate job based on the evaluation results based on different criteria(d), namely

$$X = (C_1(x), C_2(x), C_3(x) \cdots C_d(x)) \tag{1}$$

Where $C_i(x)$ is the assessed value of X based on i-th standard under current situation.

$C_1(x)$ is the urgency of the container, that

$$C_1(x) = t_d(x) - t_c \tag{2}$$

Where $t_d(x)$ is AGV's arrival time at the container, t_c is current time.The smaller $C_1(x)$ is , the more urgent the container operation will be.

$C_2(x)$ is the difference between the time when the current trolley arrives at the container and the arrival time of the AGV which completes the work fastest, namely

$$C_2(x) = t_a(v, x) - t_a(v', x) \tag{3}$$

Where$t_a(v, x)$ is the time when the AGV which is requesting the operation currently arrives at the container, and x is the empty distance from the AGV which is requesting the operation to the container. v' is the other AGV besides v which is the fastest to reach the container. When making scheduling decisions, this standard not only considers the current AGV that has already completed the operation, but also think of other AGVs that will complete their operations in the near future .

$C_3(x)$ is the empty distance from the AGV which is requesting operation to the container.

$C_4(x)$is the opposite number of the load distance of the AGV transportation operation X, which is requesting operation currently. The reason of taking the opposite number is to make the score smaller to get a better preference.

If container operation X is loaded, then $C_5(x)$ = 0; otherwise, if unloaded, then $C_5(x)$ = 1, which indicates that this strategy has a higher preference for loading than unloading. Because the most important thing is the efficiency of loading for automated container terminals.

$C_6(x)$ is the opposite number of QC's average delay, namely

$$C_6(x) = -\frac{1}{k}\sum_{i=1}^{k} D_x(X_i) \tag{4}$$

$D_x(X_i)$ is QC's operation time for the i-th container, and k is the number of containers under QC's operation so far.

$C_7(x)$ is relative remaining workload of the job X, namely

$$C_7(x)\begin{cases} -\dfrac{W_x}{2W_{avg}} & \text{Loading} \\ -\dfrac{1}{2} & \text{Otherwise} \end{cases} \tag{5}$$

As shown in figure, W_x is the number of containers in the storage yard where the current operation X is stored. W_{avg} is the average number of remaining containers in all the yards.$C_7(x)$ gives higher priority to loading containers. When the workload is evenly assigned, the value is $-\frac{1}{2}$,but for container unloading operations, the fixed value is $-\frac{1}{2}$. As a result, $C_7(x)$ can help to balance the number of containers among various yards.

Use the following normalization function to normalize the above 7 index features and treat them as values in [0,1].

$$\phi_{[l(i),\ u(i)]}(v_i) = \begin{cases} 1 & v_i > u(i) \\ 0 & v_i < l(i) \\ \dfrac{v_i - l(i)}{u(i) - l(i)} & \text{Otherwise} \end{cases} \tag{6}$$

ISSN: 2414-1895 DOI: 10.6919/ICJE.202010_6(10).0034

Where $l(i)$and $u(i)$are the empirical upper and lower limits of the i-th indicator, respectively, and v_i is the characteristic index value before the normalization of i-th indexes. The empirical boundary can be obtained by simulating multiple AGV operations which randomly generated.

2.3 Preference Function and Objective Function

For a pair of candidate jobs X_i and X_j the preference function between them can be expressed by formula (7), and the result is a real number between 0 and 1, namely

$$F: X_i^d \times X_j^d \rightarrow [0,1] \tag{7}$$

The return value close to 1 means that the first candidate job has a higher preference than the second one. When k-th different candidate jobs are given, the paired preference function can be applied to each possible combination of candidate job pairs to determine the best job. Formula (8) is used to find the sum of preference difference of candidate jobs (X_i) in the candidate job set under status θ. The job X_i with the largest value of $v(X_i|J_\theta)$ is the best job J^* in the current candidate job set J_θ, so the dispatching strategy π in the candidate job set J_θ under a given state θ is shown in formula (9).

$$v(X_i|J_\theta) = \sum\nolimits_{X_i, X_j \in J_\theta} \left(F(X_i, X_j) - F(X_j, X_i)\right) \tag{8}$$

$$\pi(J_\theta) = argmax_{X_i \in J_\theta} v(X_i|J_\theta) \tag{9}$$

There are two major goals to be achieved in the experiment: to minimize the average operating time of the QC and the empty distance of the AGV. We should measure the pros and cons of the scheduling decision in this respect, and choose the smallest $T_n$and D_n as the evaluation indicators to establish the objective function $f(n)$, supposing that the scheduling strategy has already been used to process n-th container jobs. As a result, the objective function (10) is used to measure the performance of the scheduling decision.

$$f(n) = \omega_T \times T_n + \omega_D \times D_n \tag{10}$$

Where T_n is the average operation time of processing n-th containers using QC; D_n is the average empty distance of transporting n-th containers through AGV. ω_T and ω_D are their weights respectively. The calculation function is shown as below:

$$T_n = \frac{Q}{n} \times (t_n - s) \tag{11}$$

$$D_n = \frac{1}{n} \sum\nolimits_{q \in Q} \sum\nolimits_{J \in F_{q,\ n}} e_J \tag{12}$$

Where Q is the set of QC, and t_n is the operating time of QC when the n-th job is completed. s is the start time of all jobs, and $F_{q,\ n}$ is the job set completed by QC up to time t_n. e_J is the AGV's empty distance of job J . ω_T and ω_D can vary with the relative importance of the two goals.

2.4 Sample Training and Evaluation Function

Each time a container operation is allocated to the AGV using the scheduling strategy based on the above preference function, the preference function is updated for the next round of container operation allocation. That is to say, every time a scheduling decision is completed, a short advance simulation will be performed to evaluate each candidate job. Assuming that only k-th candidate jobs are assigned, each job in k is regarded as an optimal job, and then the simulation will begin. In this process, the assignment rule ω remains unchanged. After advanced simulating k-th candidate jobs , the following objective function is used to evaluate the candidate jobs, and an evaluation matrix is obtained to find the initial job with the smallest evaluation function. While in the first AGV dispatchment, since there are no samples for learning, we will use a random strategy to obtain training samples.

The evaluation function in the advanced simulation process is as follows:

ISSN: 2414-1895 DOI: 10.6919/ICJE.202010_6(10).0034

$$f(n+k) = \omega_T \times T_{n+k} + \omega_D \times D_{n+k} \tag{13}$$

Where the calculation method of T_{n+k} and D_{n+k} is shown as below:

$$T_{n+k} = \frac{t_{n+k} - s}{min_{q \in Q} |F_{q,\ n+k}|} \tag{14}$$

$$D_{n+k} = \frac{1}{n+k} \sum_{q \in Q} \sum_{J \in F_{q,\ n+k}} e_J \tag{15}$$

The calculation method of D_{n+k} is the same as that of D_n, but T_{n+k} is obtained by calculating the average operation time of the QC with the smallest number of container operations, because if the formula (11) is used in the short-term advanced simulation, it will firstly let QC deal with those containers with a shorter operation time to get a smaller QC average operation time. However, some containers with longer processing times will not be selected until the end due to this preference, causing some delay of QC assigning those jobs. Therefore, the use of formula (14) for evaluation will promote the relatively average progress of all QC.

Firstly, find the initial job J^* with the smallest evaluation function in the evaluation matrix, and then pair the best job J^* with other candidate jobs in k to compose $(k-1)$ training samples. Because the total number of $(k-1)$training samples generated is usually not enough to help the model learn the new function reliably, the online preference learning algorithm maintains a set of recent sample pool T, where these new training samples have been accumulated in. Meanwhile, all samples in the sample pool T will learn the new preference to update the preference function for the next job assignment.

3. Online learning based on Tree-CNN

The tree convolutional neural network uses layer classifier for reference, which is composed of nodes. Like the tree in the data structure, each node has its own ID, such as parent, children, net, and LT (Labels Transform), the label corresponding to each node. As for the root node and branch node, it can help recognize a major direction of the final classification, but for the leaf node, it is more detailed and further classification. The structure of the tree convolutional neural network is shown in the Figure 2 as below. As you can see, the top is the root node of the tree. Firstly, the data is converted into a matrix and sent to the root node network for "super-classes" classification, and then according to the identified "super classes", the matrix is sent to the corresponding nodes for further classification. On the basis of different characteristics, it will classify recursively until we get the proper final result.

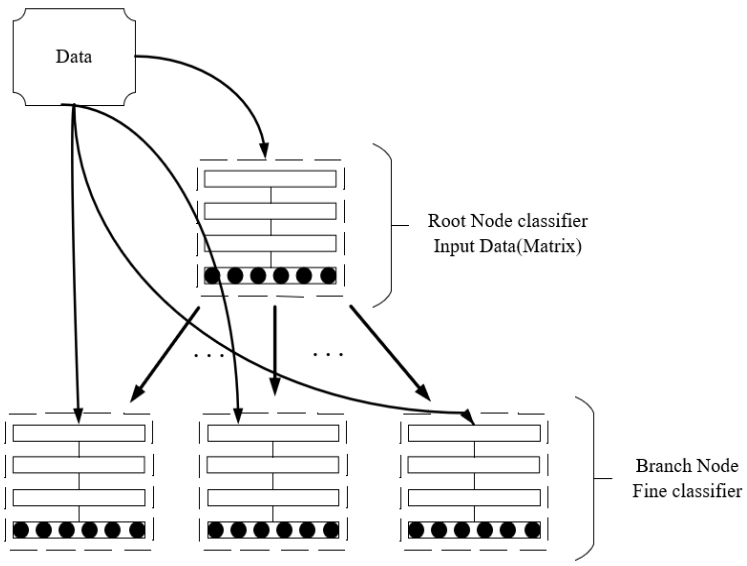

Fig. 2 structure of the tree convolutional neural network

ISSN: 2414-1895 DOI: 10.6919/ICJE.202010_6(10).0034

The preference selection among candidate jobs can be attributed to the multi-classification problem, which can be incrementally learned through the tree convolutional neural network based on multi-feature assignment strategy. The neural network is included in the nodes of the decision tree, making up a new classification model, which improves the generalization ability and reduces the time complexity. Firstly, input the matrix of 15×7 converted from the pairwise comparison of each container job to the root node. Each node is a convolutional neural network containing three layers of convolution and one layer of pooling. The output of the root node is K×M×I, where K, M and I are the number of root node's children, categories, and features of each category, The specific network structure is shown in Figure 3. Afterwards, formula (16) is used to find the average value of each type's output O_{avg}, and softmax is used in the activation function, of which output is a probability of each category. According to the position corresponding to the maximum probability, the recognition process is transferred to the next node, which will reach the leaf node finally. What the leaf node output is our final classification result, that is, the best job of AGV to be carried . In the meantime, we will continue the process mentioned above on other branch nodes. Through these operations, we will not only retain the learning of the old samples but also study the new data.

$$O_{avg}(k,\ m) = \sum_{i=1}^{I} \frac{O(k,m,i)}{I} \tag{16}$$

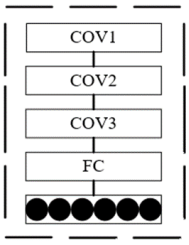

Fig. 3 The specific network structure

4. Experimental scenario and result analysis

4.1 Experimental scenario

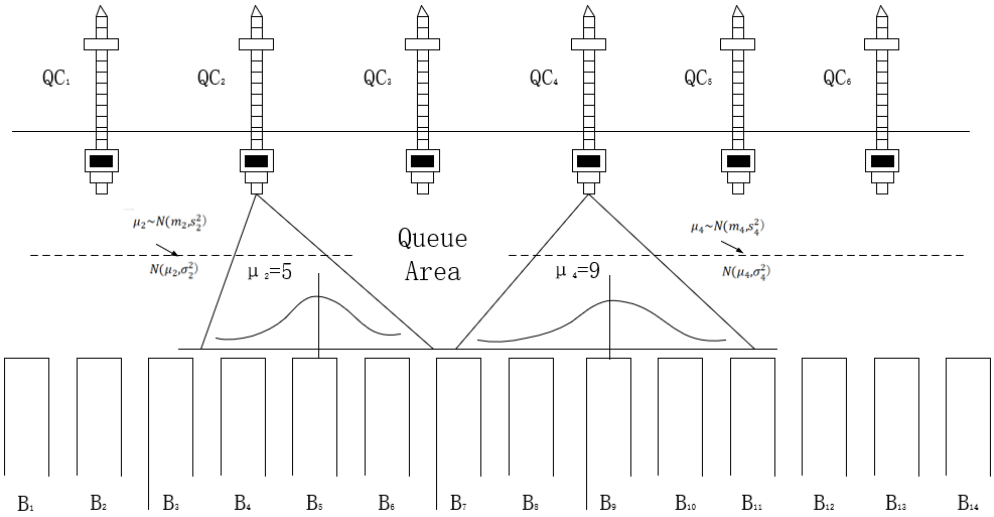

Fig. 4 Layout of an automated container terminal.

This experiment is using Python language, and the computer parameters for the experiment are Inter(R) Core(TM)i7-7700HQ CPU @ 2.80GHz 2.80 GHz, memory 16G, Windows10. The layout of the automated container terminal in the experiment is shown in Figure 4.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202010_6(10).0034

Table 2 Simulation Scenario

Number of delivery jobs	900
Number of AGVs	12,18,24,30
Number of QCs	6
Number of yard blocks	14
Number of scenarios	50 for the search and 50 for the test

Assuming there are 6 QCs, 14 container yards, 900 container operations in all. The number of simulated AGV in the experiment is 12, 18, 24 and 30, speed $v = 4m/s$ and ω_T: ω_D=1:1 .The container yard number processed by QC is following the Normal Distribution N（μ_i，σ_i^2）. For instance, when μ_4= 9, it means that the container processed by QC_4 is stored near the yard B_9. With the increases or decreases with σ_4 , the range of container stored around B_9 is increasing or decreasing. Therefore, we can simulate different experimental scene by changing μ_i and σ_i. There are 900 containers included in the experiment,which are distributed widely in different yards.What's more, the load distance of the AGV is uneven.

4.2 Experimental analysis

4.2.1 Experiment 1: Compare the performance of the three algorithms under different AGV numbers

Table 3-Table 6 show the comparative experimental results of different solving algorithms when the number of AGV are 12, 18, 24 and 30. The first and second columns respectively represent the average operating time of each QC and the empty distance of the AGV during processing operations, and the third column represents the training time. They are the average values obtained under 50 different test scenarios.

Table 3 Performances of dispatching strategies by different methods with 12 AGVs

	QC(s)	AGV(m)	Time(s)
Tree-CNN	195.30	397.66	6.58
CNN	192.05	382.05	1.89
BP	182.44	432.04	2.69

Table 4 Performances of dispatching strategies by different methods with 18 AGVs

	QC(s)	AGV(m)	Time(s)
Tree-CNN	169.05	374.98	7.46
CNN	173.35	362.44	1.89
BP	169.82	417.32	3.57

Table 5 Performances of dispatching strategies by different methods with 24 AGVs

	QC(s)	AGV(m)	Time(s)
Tree-CNN	147.54	356.42	9.57
CNN	161.58	354.43	2.47
BP	157.22	406.08	4.96

Table 6 Performances of dispatching strategies by different methods with 30 AGVs

	QC(s)	AGV(m)	Time(s)
Tree-CNN	138.92	346.27	14.38
CNN	157.36	350.09	5.26
BP	155.21	403.87	7.85

ISSN: 2414-1895 DOI: 10.6919/ICJE.202010_6(10).0034

(1) Combined Table 3 and Table 4, when the number of AGV increases from 12 to 18, the average QC operation time is reduced by 15.3%, and the empty distance of AGV is reduced by 5.7% when using Tree-CNN to solve. Under the circumstance of using CNN and BP in the same period, the average operation time of QC was reduced by 9.7% and 8.5%, and the AGV's empty distance was reduced by 5.0% and 3.4%, respectively. When the number of AGV is 18, the average operation time of QC under Tree-CNN is the shortest one within the three solving methods. The empty distance of the AGV is 3% longer than that under CNN.

(2) Combined Table 4 and Table 5, when the number of AGV raises from 18 to 24, the average QC operation time is reduced by 9.2%, and the AGV empty distance is reduced by 5.0% when using Tree-CNN. In the situation of using CNN and BP in the same period, the average operation time of QC was reduced by 6.7% and 5.6%, respectively, and the empty distance of AGV was reduced by 2.3% and 2.5%, respectively. At this time, under Tree-CNN, the average operation time of QC is still the shortest one under the three solving methods, and the difference between the empty distance of the AGV and the empty distance of the AGV obtained under CNN is only 0.56%. At present, Tree-CNN's advantages in QC average operating time and empty distance of AGV gradually become obvious.

(3) Combined Table 5 and Table 6, when the number of AGV increases from 24 to 30, the average QC operation time is reduced by 6.5%, and the AGV empty distance is reduced by 2.9% when using Tree-CNN. In the case of using CNN and BP in the same period, the average operation time of QC was reduced by 2.5% and 1.2%, respectively, and the AGV's empty distance was reduced by 1.2% and 0.6%, respectively. At this moment, the QC average operating time and the AGV's empty distance obtained by using Tree-CNN are the shortest under the three methods.

Compare the training time of the three methods, because Tree-CNN is using neural network fusion technology (similar to random forest), through including the neural network in the nodes of the decision tree, where each node adds three fully connected layers. Therefore, the training time under Tree-CNN gradually increases. When the number of AGV is 30, the training time is 14.38s. The average time of the scheduling interval set in the simulation setting is 15s. Compared with the increase ratio of the other two algorithms, in rare cases, continuous scheduling is required in a short interval, so the time is acceptable. Otherwise, the learning process will be aborted and changed according to the last dispatching strategy for scheduling.

In summary, as the number of AGV increases and the workload increases, compared with CNN and BP, the advantages of using Tree-CNN to solve will become more evident, and when the number of AGV reaches to 30, within the average time in the effective scheduling interval, Tree-CNN will achieve the best QC average operating time and AGV's empty distance.

4.2.2 Experiment 2: Analyze the impact of weight on performance

In experiment 1, the weight ω_T: ω_D =1:1. However, the ratio of ω_T: ω_D should actually be determined by the operator of the automated terminal according to their own operational goals. Therefore, the performance results under two extreme strategies are tested by changing the weight ratio: one with the shortest average QC operation time, and the other one has the shortest empty distance of AGV. The experimental results are shown in Figures 5, 6, 7, and 8. Three different shapes represent their own solution methods. Each color is set with 6 different weight ratios from right to left, where ω_T: ω_D are 1:1, 10:1, 50:1, 100:1, 500: 1 and 1000:1.

As shown in Figure 5, when the number of AGV is 12, no matter what the weight ratio is, the average QC operation time obtained by BP is the shortest one, while the empty distance of the AGV obtained by CNN is the shortest one. Therefore, if only pursue the goal of the minimum QC average operating time, BP can be used to solve the problem. If only consider the minimum empty distance of the AGV, CNN will be the best choice.

As seen in Figure 6, when the number of AGV raises to 18, as the weight ratio increases, the performance of Tree-CNN gradually catches up with BP in reducing the average operation time of QC, but at the same time, compared with BP, the empty distance of AGV is reduced by about 8.2%, so if you consider the minimum QC average operating time, you can use Tree-CNN. Taking the empty

ISSN: 2414-1895 DOI: 10.6919/ICJE.202010_6(10).0034

distance of the AGV into account, when ω_T:ω_D is equal to 1:1, the empty distance of the AGV obtained by CNN is about 3.3% less than that of Tree-CNN. Besides, when ω_T:ω_D is equal to 100:1, compared with Tree-CNN, the empty distance of AGV with CNN is shortened to about 1.5%. Namely, as the weight ratio increases, the difference between the results obtained under the final use of CNN and Tree-CNN is only 0.3%, indicating that Tree-CNN can better adapt to this Extreme assignment strategy, and always maintain the best performance in reducing QC average operating time.

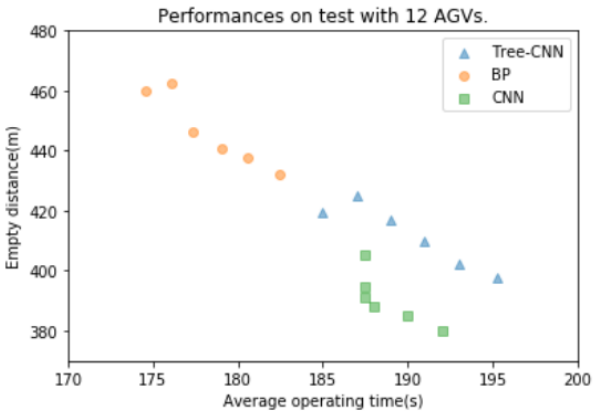

Fig. 5 Performances on the test with 12 AGVs

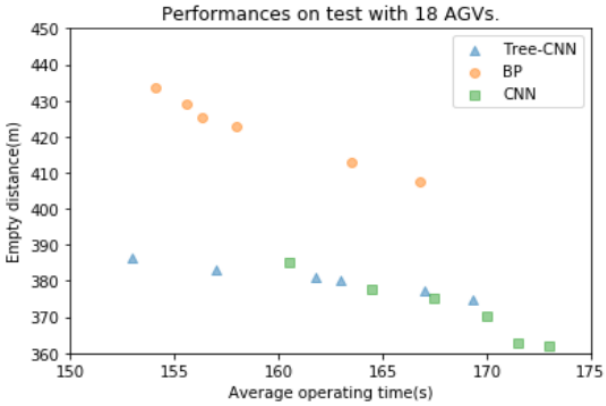

Fig. 6 Performances on the test with 18 AGVs

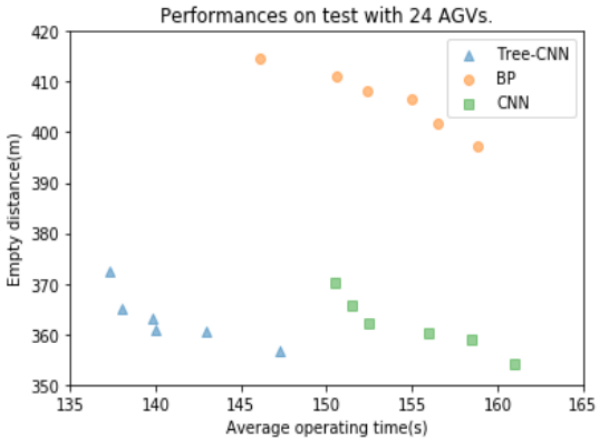

Fig. 7 Performances on the test with 24AGVs

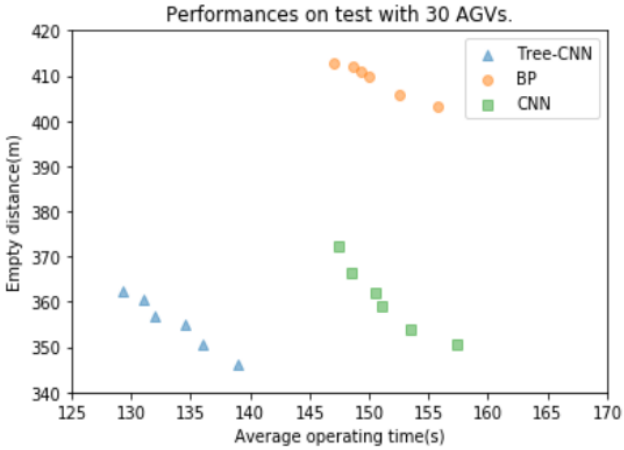

Fig. 8Performances on the test with 30 AGVs

As shown in Figure 7, when the number of AGV increases to 24, regardless of the weight ratio, Tree-CNN shows the best performance in terms of QC average operating time and AGV's empty distance, especially in QC average operating time. Compared with CNN and BP, the average operation time of QC is reduced by 8.6% and 6.6% respectively when using Tree-CNN. At present, whether considering the shortest QC average operating time or minimizing the empty distance of the AGV, Tree-CNN will be your top choice.

As you can see in Figure 8, when the number of AGV increases to 30, Tree-CNN still maintains the best performance in reducing QC average operating time and AGV's empty distance. Nevertheless, with the weight ratio increases, the contribution of Tree-CNN, CNN and BP in reducing the average QC operation time and the AGV empty distance decreases, indicating that the number of AGV at this time has a higher impact on the results than the weight ratio.

In conclusion, when the number of AGV are 12, 18 and 24, the weight has a greater impact on the performance results of the dispatching strategy. As the weight ratio increases, Tree-CNN can better be used to the dispatching strategy and keep strengths in reducing QC The average operating time and the empty distance of AGV. When the number of AGV increases to 30, the weight change has

ISSN: 2414-1895 DOI: 10.6919/ICJE.202010_6(10).0034

less impact on the results. It can be speculated that the multi-feature dispatching strategy based on Tree-CNN is suitable for which number of AGV is between 24 and 30.

5. Conclusion

This paper designs a multi-feature dispatching rule based on container operations, with the goal of minimizing QC average operating time and AGV's empty distance. Use neural network fusion technology, that is, Tree-CNN, for online learning based on multi-feature dispatching strategy, and change the dispatching strategy by updating the preference function. Compared with ordinary CNN, Tree-CNN improves the generalization ability of new data. In addition, learning decision tree type neurons incrementally, it doesn't need to be rebuilt while new training data is encountered, namely, it can store the memory of old data. However, as more networks are built, training time will become longer. The average time of the scheduling interval set in the simulation settings is 15s. In rare cases, continuous scheduling is required in a short interval, so the time is acceptable. Otherwise, the learning process will be stopped. The experiment compares the results of learning dispatching strategy using three algorithms including Tree-CNN, CNN and BP under the situation where the number of AGV is equal to 12-30. When the number of AGV increases to 24, under Tree-CNN, the average operation time of QC is still the shortest one under the three solving methods, and the difference between the empty distance of the AGV and the empty distance of the AGV obtained under CNN is only 0.56%. At this time, Tree-CNN gradually shows its advantages in reducing the average QC operation time and the empty distance of the AGV. When the number of AGV is equal to 30, within the effective average time of the scheduling interval, Tree-CNN achieves the best performance in the respect of QC average operating time and AGV's empty distance, which verifies that the proposed multi-feature dispatching strategy can effectively improve the production efficiency of the terminal. Finally, by changing the weight ratio of the objective function, the performance results under the two extreme strategies are compared. When the number of AGV are 12, 18, and 24, the weight has a greater impact on the performance results of the dispatching strategy. With the weight ratio increases, Tree-CNN can better adapt to the assignment strategy and maintain the advantage in reducing the average QC operation time and the AGV's empty distance. When the number of AGV increases to 30, the weight change will have less impact on the results. It can be speculated that, multi-feature dispatching strategy based on Tree- CNN is applicable to the number of AGV between 24 and 30.

Acknowledgements

This research was supported by the Shanghai Pujiang Program(16PJC043).

References

[1] ZHANG Yaqi, YANG Bin, HU Zhihua, TIAN Maojin. Research of AGV charging and job integrated scheduling at automated container terminal [J]. Computer Engineering and Applications, 2017, 53(18): 257-262+270.

[2] WU QiPing, JIN Yaping, REN Ping. Present situation and developing trend of AGV key technology [J]. Manufacturing automation, 2013,35(10):106-109+121.

[3] Dang Q V,Nielsen I, Steger-Jensen K,et al. Dispatching a single mobile robot for part-feeding tasks of production lines[J]. Journal of Intelligent Manufacturing, 2014,25(6):1271-1287.

[4] MA Yue-hui HU Zhi-hua.The dispatching and scheduling problems of AGVs at automated container terminals under uncertainty conditions[J] Journal of Guangxi University(Natural Science Edition), 2016, 41(2):589-597.

[5] Wu Yaohua,XIAO Jiwei, Gao Weifeng. AGV task and route scheduling problem solution[J], Manufacturing automation, 2000(11):40-42.

[6] Ci Chen, Lifeng Xi, Binghai Zhou, et al. A multiple-criteria real-time scheduling approach for multiple-load carriers subject to LIFO loading constraints[J]. International Journal of Production Research, 2011,49(16): 4787-4806.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202010_6(10).0034

[7] Xie Mingli Hu Zhanqi Ma Ning, Research on scheduling strategy of flexible manufacturing system based on maximum entropy neural network algorithm[J/OL]. Application Research of Computers. 2018, 35(12):1-6.

[8] Akyol D E, Bayhan G M. Multi-machine earliness and tardiness scheduling problem: an interconnected neural network approach[J]. International Journal of Advanced Manufacturing Technology, 2008,37(5-6): 576-588.

[9] ChoeR, Kim J, Ryu K R. Multi-objective optimization of dispatching strategies for situation-adaptive AGV operation in an automated container terminal[C], Proceedings of the 2013 Research in Adaptive and Convergent Systems October 2013 Pages 1–6

[10] Choe, R.et al. 2011. Queue-based local scheduling and global coordination for real-time operation control in a container terminal. Journal of Intelligent Manufacturing. 23, 6(Jul. 2011), 2179--2192

[11] XIN Jian-bin, NEGENBORN R R, CORMAN F, et al. Control of interacting machines in automated container terminal using a sequential planning approach for collision avoidance[J]. Transportation Research Part C: Emerging Technologies.2015:60,377-396.

[12] Yeou-Ren Shiue, Ken-Chuan Lee, Chao-Ton Su. Real-time scheduling for a smart factory using a reinforcement learning[J]. Computers & Industrial Engineering. 2018.