A REINFORCEMENT LEARNING FOR MARSHALING OF FREIGHT CARS IN A TRAIN BASED ON THE PROCESSING TIME

Yoichi Hirashima
Faculty of Information Science and Technology,
Osaka Institute of Technology,
1-79-1, Kita-yama, Hirakata City, Osaka, 573-0196, Japan
hirash-y@is.iot.ac.jp

Abstract

In this paper a new method for generating marshaling plan of freight cars in a train is proposed. In the proposed method, marshaling plans based on the processing time can be obtained by a reinforcement learning system. In order to evaluate the processing time, the total transfer distance of a locomotive and the total movement counts of freight cars are simultaneously considered. The order of movements of freight cars, the position for each removed car, the layout of cars in a train and the number of cars to be moved are simultaneously optimized to achieve minimization of the total processing time for obtaining the desired layout of freight cars for an outbound train. Initially, freight cars are located in a freight yard by the random layout, and they are moved and lined into a main track in a certain desired order in order to assemble an outbound train. Learning algorithm in the proposed method is based on the Q-Learning, and is applied to reflect the processing time that are used to achieve one of the desired layouts in the main track. After adequate autonomous learning, the optimum marshaling plan can be obtained by selecting a series of movements of freight cars that has the best evaluation.

Keywords: Scheduling, Container Transfer Problem, Q-Learning, Freight train, Marshaling

1. INTRODUCTION

Train marshaling operation at freight yard is required to joint several rail transports, or different modes of transportation including rail. Transporting goods are carried in containers, each of which is loaded on a freight car. A freight train consists of several freight cars, and each car has its own destination. Thus, the train driven by a locomotive travels several destinations disjointing corresponding freight cars at each freight station. In addition, since freight trains can transport goods only between railway stations, modal shifts are required for delivering them to area that has no railway. In intermodal transports from the road and the rail, containers carried into the station are loaded on freight cars in the arriving order. The initial layout of freight cars is thus random. For efficient shift, the desirable layout should be determined considering destination of container. Then, freight cars must be rearranged before jointing to the freight train. In general, the rearrangement process is conducted in a freight yard that consists of a main track and several sub-tracks. Freight cars are initially placed on sub tracks, rearranged, and lined into the main track. This series of operation is called marshaling, and several methods to solve the marshaling problem have been proposed (Blasum et al. 2000, Jacob et al. 2007). Also, many similar problems are treated by mathematical programming and genetic algorithm (Kroon et al. 2008, Tomii and Jian 2000, He et al. 2000, Dahlhaus et al. 2000), and some analyses are conducted for computational complexities (Dahlhaus et al. 2000, Eggemont 2009). However, these methods do not consider the processing time for each transfer movement of freight car that is moved by a locomotive.

In this paper a new method for generating marshaling plan of freight cars in a train is proposed. In the proposed method, marshaling plans based on the processing time can be obtained by a reinforcement learning system. A movement of a freight car consists of 4 elements: 1. moving a locomotive to the car to be transferred, 2. jointing cars with the locomotive, 3. transferring cars to their new position by the locomotive, and 4. disjoining the cars from the locomotive. The processing times for elements 1. and 3. are determined by the transfer distance of the locomotive, the weight of the train, and the performance of the locomotive. The total processing time for elements 1. and 3. is determined by the number of movements of freight cars. Thus, the transfer distance of the locomotive and the number of movements of freight cars are simultaneously considered, and used to evaluate and minimize the processing time of marshaling for obtaining the desired layout of freight cars for an outbound train. The total processing time of marshaling is considered by using a weighted cost of a transfer distance of the locomotive and the number of movements of freight cars. Then, the order of movements of freight cars, the position for each removed car, the layout of cars in a train and the number of cars to be moved are simultaneously optimized to achieve minimization of the total processing time. The original desired layout of freight cars in the main track is derived based on the destination
of freight cars. In the proposed method, several desirable layouts of freight cars in the main track are generated from the original one, and the optimal layout that can achieve the smallest processing time of marshaling is obtained by autonomous learning. Simultaneously, the optimal sequence of car-movements as well as the number of freight cars that can achieve the desired layout is obtained by autonomous learning. Also, the feature is considered in the learning algorithm, so that, at each arrangement on sub track, an evaluation value represents the smallest processing time of marshaling to achieve the best layout on the main track. The learning algorithm is derived based on the Q-Learning (Watkins and Dayan 1992), which is known as one of the well established realization algorithm of the reinforcement learning.

In the learning algorithm, the state is defined by using a layout of freight cars, the car to be moved, the number of cars to be moved, and the destination of the removed car. An evaluation value called Q-value is assigned to each state, and the evaluation value is calculated by several update rules based on the Q-Learning algorithm. In the learning process, a Q-value in a certain update rule is referred from another update rule, in accordance with the state transition. Then, the Q-value is discounted according to the transfer distance of the locomotive. Consequently, Q-values at each state represent the total processing time of marshaling to achieve the best layout from the state. Moreover, in the proposed method, only referred Q-values are stored by using table look-up technique, and the table is dynamically constructed by binary tree in order to obtain the best solution with feasible memory space. In order to show effectiveness of the proposed method, computer simulations are conducted for several methods.

2. Problem Description

![Diagram of freight yard](image)

Fig. 1  Freight yard

The yard consists of 1 main track and m sub tracks. Define k as the number of freight cars placed on the sub tracks, and they are carried to the main track by the desirable order based on their destination. In the yard, a locomotive moves freight cars from sub track to sub track or from sub track to main track. The movement of freight cars from sub track to sub track is called removal, and the car-movement from sub track to main track is called rearrangement. For simplicity, the maximum number of freight cars that each sub track can have is assumed to be n, the ith car is recognized by a unique symbol c_i (i = 1,...,k). Fig.1 shows the outline of freight yard in the case k = 30; m = n = 6. In the figure, track Tm denotes the main track, and other tracks [1], [2], [3], [4], [5], [6] are sub tracks. The main track is linked with sub tracks by a joint track, which is used for moving cars between sub tracks, or for moving them from a sub track to the main track. In the figure, freight cars are moved from sub tracks, and lined in the main track by the descending order, that is, rearrangement starts with c30 and finishes with c1. When the locomotive L moves a certain car, other cars located between the locomotive and the car to be moved must be removed to other sub tracks. This operation is called removal. Then, if k ≤ nm − (n − 1) is satisfied for saving adequate space to conduct removal process, every car can be rearranged to the main track.

In each sub track, positions of cars are defined by n rows. Every position has unique position number represented by nm integers. Fig.2 shows an example of position index for k = 30, m = n = 6 and the layout of cars for fig.1. In Fig.2, the position “[a][1]” that is located at row “[a]” in the sub track “[1]” has the position number 1, and the position “[f][6]” has the position number 36. For unified representation of layout of car in sub tracks, cars are placed from the row “[a]” in every track, and newly placed car is jointed with the adjacent freight car. In the figure, in order to rearrange c25, other cars c21, c22, c23, c24, and c20 that are located in front of c25 have to be removed to other sub tracks. Then, since k ≤ nm − (n − 1) is satisfied, the cars located in front of c25 can be removed, and c25 can be rearranged to the main track.

In the freight yard, define x_i (1 ≤ x_i ≤ nm, i = 1,...,k) as the position of the car c_i, and s = [x_1, ..., x_k] as the state vector of the sub tracks. For example, in Fig.2, the state is represented by s = [1, 7, 13, 19, 25, 31, 2, 8, 14, 20, 26, 32, 3, 9, 15, 21, 4, 10, 5, 36, 12, 18, 24, 30, 6]. A trial of the rearrange process starts with the initial layout, rearranging freight cars according to the desirable layout in the main track, and finishes when all the cars are rearranged to the main track.

3. Desired Layout In The Main Track

In the main track, freight cars that have the same destination are placed at the neighboring positions. In this case, removal operations of these cars are not required at the destination regardless of layouts of these cars. In order to consider this feature in the desired layout in the main track, a group is organized by cars that have the same destination, and these cars can be placed at any positions in the group.
Then, for each destination, make a corresponding group, and the order of groups lined in the main track is predetermined by destinations. This feature yields several desirable layouts in the main track. Fig. 3 depicts examples of desirable layouts of cars and the desired layout of groups in the main track. In the figure, freight cars $c_1, c_2, \ldots, c_6$ bound for the destination 1, make group 1, and freight cars $c_7, c_8, \ldots, c_{18}$ bound for the destination 2, make group 2. In the figure, freight cars $c_{25}$, $c_{26}, \ldots, c_{30}$ for the destination 3, make group 3, and freight cars $c_{31}$, $c_{32}, \ldots, c_{36}$ for the destination 4, make group 4. Groups $1, 2, 3, 4$ are lined by ascending order in the main track, which make a desirable layout. In the figure, examples of layout in group 1 are in the dashed square.

![Fig. 3 Example of groups](image)

### 4. DIRECT REARRANGEMENT

When rearranging a car that has no car to be removed on it is exist, its rearrangement precede any removals. In the case that several cars can be rearranged without a removal, rearrangements are repeated until all the candidates for rearrangement require at least one removal. If several candidates for rearrangement require no removal, the order of selection is random, because any orders satisfy the desirable layout of groups in the main track. In this case, the arrangement of cars in sub tracks obtained after rearrangements is unique, so that the movement counts of cars has no correlation with rearrangement orders of cars that require no removal. This operation is called direct rearrangement. When a car in a certain sub track can be rearrange directly to the main track and when several cars located adjacent positions in the same sub track satisfy the layout of group in main track, they are jointed and applied direct rearrangement.

Fig. 4 shows an example of arrangement in sub tracks existing candidates for rearranging cars that require no removal. At the top of figure, from the left side, a desired layout of cars and groups, the initial layout of cars in sub tracks, and the position index in sub tracks are depicted for $k = 9, m = n = 4$. In the figure, $c_1, c_2, c_3, c_4$ are in group 1, $c_5, c_6, c_7, c_8$ are in group 2, and group 1, must be rearranged first to the main track. In each group, any layouts of cars can be acceptable. In both cases, $c_2$ in step 1 and $c_3$ in step 3 are applied the direct rearrangement. Also, in step 4, 3 cars $c_1, c_4, c_5$ located adjacent positions are jointed and moved to the main track by a direct rearrangement operation. In addition, at step 5 in Case 2, cars in group 2 and group 3 are moved by a direct rearrangement, since the positions of $c_7, c_8, c_9, c_9$ are satisfied the desired layout of groups in the main track.

In “Case 1” of the example, the rearrangement order of cars that require no removal is $c_1, c_2, c_3, c_4$, and in “Case 2”, the order is $c_3, c_2, c_1, c_4$. Although 2 cases have different orders of rearrangement, the arrangements of cars in sub tracks and the numbers of movements of cars have no difference.

![Fig. 4 Direct rearrangements](image)

### 5. REARRANGEMENT PROCESS

The rearrangement process for cars consists of following 5 operations:

1. rearrangement for all the cars that can apply the direct rearrangement into the main track,
2. selection of a freight car to be rearranged into the main track,
3. selection of a removal destination of cars located between the locomotive and the freight car selected in (2),
4. removal of the cars to the selected sub track,
5. rearrangement of the selected car to the main track.

These operations are repeated until one of desirable layouts is achieved in the main track, and a series of operations from the initial state to the desirable layout is defined as a trial.

In the operation (2), each group has the predetermined position in the main track. The car to be rearranged is defined as $c_T$, and candidates of $c_T$ can be determined by excluding freight cars that have already rearranged to the main track. These candidates must belong to the same group that is determined uniquely by the desired layout of groups in the main track and the number of rearranged cars.

Now, define $r$ as the number of groups, $g_l$ as the number of freight cars in group $l$ $(1 \leq l \leq r)$, and $u_j$ $(1 \leq j \leq g_l)$ as candidates of $c_T$.

In the operation (3), the removal destination of car located on the car to be rearranged is defined as $r_M$. Then, defining
when
as candidates of \( r_M \), excluding
the
sub track that has the car to be removed, and the number of candidates is \( m - 1 \).
In the operation (4), defining \( p_S \) as the number of removal cars required to rearrange \( c_T \), and defining \( p_d \) as the number of removal cars that can be located on the sub track selected in the operation (3), the candidate numbers of cars to be moved are determined by \( u_j, 2m \leq j \leq \frac{3}{2}m + \min\{p_S, p_d\} - 1 \).
In both cases of Fig.4, the direct rearrangement is conducted for \( c_2 \) at step1, and the selection of \( c_T \) conducted at step2, candidates are \( u_1 = [1], u_2 = [4] \), that is, sub tracks where cars in group1 are located at the top. \( u_3, u_4 \) are excluded from candidates. Then, \( u_2 = [4] \) is selected as \( c_T \). Candidates for the location of \( c_T \) are \( u_5 = [1], u_6 = [2], u_7 = [3] \), sub tracks \([1],[2], \) and \([3] \). In Case1, \( u_6 = [2] \) is selected as \( c_T \), and in Case2, \( u_7 = [3] \) is selected. After direct rearrangements of \( c_2 \) at step3 and \( c_1, c_4, c_5 \) at step4, the marshaling process is finished at step5 in Case2, whereas Case1 requires one more step to in order to finish the process. Therefore, the layout of cars and groups in the main track, the number of cars to be moved, the location of the car to be rearranged and the order of rearrangement affect the total movement counts of cars as well as the total transfer distance of locomotive.

6. PROCESSING TIME FOR A MOVEMENT OF LOCOMOTIVE

6.1 Transfer distance of locomotive

When a locomotive transfer freight cars, the process of the unit transition is as follows: (E1), starts without freight cars, and reaches to the joint track, (E2) restart in reverse direction to the target car to be moved, (E3), joins them, (E4) pull out them to the joint track, (E5) restart in reverse direction, and transfers them to the indicated location, and (E6) disjoins them from the locomotive. Then, the transfer distance of locomotive in (E1), (E2), (E4) and (E5) are defined as \( D_1, D_2, D_3 \) and \( D_4 \), respectively. Also, define the unit distance of a movement for cars in each sub track as \( D_{min} \), the length of joint track between adjacent sub tracks, or, sub track and main track as \( D_{min} \). The location of the locomotive at the end of above process is the start location of the next movement process of the selected car. Also, the initial position of the locomotive is located on the joint track nearest to the main track.

Fig.5 shows an example of transfer distance. In the figure, \( m = n = 6, D_{min} = D_{min} = 1, k = 18, \) (a) is position index, and (b) depicts movements of locomotive and freight car. Also, the locomotive starts from position 8, the target is located on the position 18, the destination of the target is 4, and the number of cars to be moved is 2. Since the locomotive moves without freight cars from 8 to 24, the transfer distance is \( D_1 + D_2 = 12 (D_1 = 5, D_2 = 7) \), whereas it moves from 24 to 16 with 2 freight cars, and the transfer distance is \( D_3 + D_4 = 13 (D_3 = 7, D_4 = 6) \).
where $\alpha$ is the learning rate, $R$ is the reward that is given when one of desirable layout is achieved, and $\gamma$ is the discount factor that is used to reflect the processing time of the marshaling and calculated by the following equation.

$$\gamma = \frac{\delta f_{\text{max}} - \beta h_{\text{U}}}{f_{\text{max}}}, 0 < \beta < 1, 0 < \delta < 1$$ (5)

Propagating Q-values by using eqs.(2)-(5), Q-values are discounted according to the processing time of marshaling. In other words, by selecting the removal destination that has the largest Q-value, the processing time of the marshaling can be reduced.

In the learning stages, each $u_j (1 \leq j \leq 2m + \min\{p_S, p_D\} - 1)$ is selected by the soft-max action selection method (Sutton and Barto 1999). Probability $P$ for selection of each candidate is calculated by

$$Q_i(s, u_j) = \frac{Q_i(s, u_j) - \min_u Q_i(s, u_j)}{\max_u Q_i(s, u_j) - \min_u Q_i(s, u_j)}$$ (6)

$$P(s_i, u_j) = \frac{\exp(\hat{Q}(s_j, u_j) / T)}{\sum_{u \in U_j} \exp(\hat{Q}(s_j, u) / T)}, \quad (i = 1, 2, 3).$$ (7)

In the addressed problem, $Q_1, Q_2, Q_3$ become smaller when the number of discounts becomes larger. Then, for complex problems, the difference between probabilities in candidate selection remain small at the initial state and large at final state before achieving desired layout, even after repetitive learning. In this case, the obtained evaluation does not contribute to selections in initial stage of marshaling process, and search movements to find better solution is not selected in final stage. To conquer this drawback, $Q_1, Q_2, Q_3$ are normalized by eq.(6), and the thermo constant $T$ is switched from $T_1$ to $T_2$ ($T_1 > T_2$) when the following condition is satisfied:

$$[\text{The count of } Q_i(s_j, u_j)] > \eta, \ s.t. \ Q_i(s_j, u_j) > 0, \quad 0 < \eta \leq \text{[the number of candidates for } u_j]\)$$ (8)

where $\eta$ is the threshold to judge the progress of learning.

The proposed learning algorithm can be summarized as follows:

1. **Initialize all the Q-values as 0**
2. **When no cars are placed on candidates of $c_T$, all of them are rearranged**
   - Update corresponding $Q_3(s_3, p_m)$ by eq.(4)
   - Store $s_3, c_T$
3. **If no cars are in sub tracks, go to 9, otherwise go to 4**
4. **Determine $c_T$ among the candidates by roulette selection (probabilities are calculated by eq. (7)).**
   - Put reward as $R = 0$,
   - Update the corresponding $Q_3(s_3, p_m)$ by eq.(4)
   - Store $s_1, c_T$

8. **NUMERICAL EXAMPLES**

Computer simulations are conducted for $m = 12, n = 6, k = 36$ and the total processing time and the learning performances of following 3 methods are compared: (A) proposed method that evaluates the processing time of movement of the locomotive, and uses 2 thermo constants $T_1, T_2$ with normalized evaluation values, (B) a method that evaluates the transfer distance of the locomotive, uses 2 thermo constants with normalized evaluation values, (C) a method that evaluates the number of movements of freight cars (Hirashima 2010), and uses 2 thermo constants with normalized evaluation values.

The initial arrangement of cars in sub tracks is described in Fig.7, desirable layout considering groups in the main track is depicted in Fig.8. In this case, the arrangement order of groups is group1, group2, group3, group4. Cars $c_1, \ldots, c_9$ are in group1, $c_{10}, \ldots, c_{18}$ are in group2, $c_{19}, \ldots, c_{27}$ are in group3, and $c_{28}, \ldots, c_{36}$ are in group4. Other parameters are set as $\alpha = 0.9, \beta = 0.2, \delta = 0.9, R = 1.0, \eta = 0.95, T_1 = 0.1, T_2 = 0.05$. 

![Fig. 6 Flowchart of the learning algorithm](image-url)
The locomotive assumed to accelerate and decelerate the train with the constant force $100 \times 10^3 \text{N}$, and to be $100 \times 10^3 \text{kg}$ in weight. Also, all the freight cars have the same weight, $50 \times 10^3 \text{kg}$. The locomotive and freight cars assumed to have the same length, and $D_{\text{min}} = D_{\text{max}} = 20 \text{m}$. The velocity of the locomotive is limited to no more than 10m/s. Then, the locomotive accelerates the train until the velocity arrives 10m/s, keeps the velocity, and decelerates until the train stops within the indicated distance. When the velocity does not arrive 10m/s at the half way point, the locomotive starts to decelerate immediately. Then, $t_{\text{max}} = 462$.

Fig. 9 show the results. In Fig.9, horizontal axis expresses the number of trials and the vertical axis expresses the minimum processing time of whole marshaling process to achieve a desirable layout found in the past trials. Each result is averaged over 20 independent simulations. In Fig.9, as the number of trials increases, the total processing time of marshaling reduces, and method (A) derives solutions that require smaller processing time of marshaling as compared to methods (B) and (C). Since method (A) evaluate the processing time directly, the method is effective for reducing the total processing time of marshaling.

Fig. 7 Initial layout

Fig. 8 Yard setting

Fig. 9 Comparison of learning performances

9. CONCLUSIONS

A new scheduling method has been proposed in order to rearrange and line cars in the desirable order onto the main track. The learning algorithm of the proposed method is derived based on the reinforcement learning, considering the total processing time of marshaling. In order to reduce the total processing time of marshaling, the proposed method learns the number of cars to be moved, as well as the layout of main track, the rearrangement order of cars, and the removal destination of cars, simultaneously. In computer simulations, learning performance of the proposed method has been improved by using normalized evaluation and switching thermo constants in accordance with the progress of learning.

References


