AN OPTIMIZATION APPROACH TO RAILWAY TRACK ALLOCATION CONSIDERING THE TRAIN-SET ROUTINGS

BUM HWAN PARK
Dept. of Railroad Management and Logistics, Korea National University of Transportation, Korea

ABSTRACT

This paper deals with an optimization approach to railway track allocation, which is one of the most important decision problems in the presence of multiple train operating companies (TOCs). In fact there has been deep discussion about how to boost the competition environment in Korean railway since the functional reform in 2004, which at last resulted in introducing a new entrant to high-speed railway passenger transportation market. Finally, in August 2016, two operating companies will compete on the major routes in the Korean high-speed railway network. The infra manager, KR Network, who is responsible for allocating the slots, has been developing their own allocation procedure which partly uses an optimization model for adjusting the times of requested train-paths. But one of the TOCs' concerns with respect to the adjustment is that their train-set routing plan could be in disorder by the adjustment of the arrival/departure times. Assuming TOCs submit their routing plan as well as their desired train-paths, we present an optimization model and algorithm for track allocation problem, considering the routing plan requested by TOCs. The model is developed on a time-space network, where a train-path can be described as the sequence of the arcs. Based on the network, we developed an column-generation approach to dynamically generate the promising train-paths for each requested one so as to maximize the total profit while preventing the routing plans from disrupting by means of setting up the arcs only among the two successive train-paths in the routing plan. Also we present the experimental results applied to the Korean high-speed railway network.

Keywords : track allocation, column generation, train-set routing.

1 INTRODUCTION

After the functional reform of the Korean railway to separate the train operating company (TOC) from the infrastructure manager in 2004, the Korean railway has been trying to foster the competitive environment in the railway passenger transportation market. At last, the efforts resulted in introducing a new entrant to high-speed railway passenger transportation market. Finally, in August 2016, two TOCs, KORAIL and SR, will compete on the major routes in the Korean high-speed railway network.

The infrastructure manager, the KR Network, has been developing its own procedure to allocate the tracks to the operating companies. The procedure is based on an optimization model to find the conflict-free timetable by adjusting arrival/departure times of each train submitted by the operating companies. But one of the operating companies' biggest concerns with respect to the adjustment is that their train-set routing plan could be in disorder by the adjustment of the arrival/departure times. In fact, the operating companies try to convince the infra manager to respect the train-set routing plan. However, the consideration of the routing plans could make the adjustment procedure more difficult.

Our study deals with an optimization approach to railway track allocation, respecting the routing plans considered by the TOCs.

This paper is part of the proceedings of the 15th International Conference on Railway Engineering Design and Operation (COMPRAIL)
www.witconferences.com

ISSN: 2058-8305 (paper format), ISSN: 2058-8313 (online), http://www.witpress.com/journals
DOI: 10.2495/TDI-V1-N3-452-459

In fact, the track allocation problems, which is in essence a combination of timetabling problem and bidding procedure, have received attention in a few literature: see, for example, Borndörfer et al. [1], Nilsson [2], Brewer and Plott [3] and Parkes and Ungar [4]. These works need a variant of timetabling problem which is called a *winner determination problem* in the context of *auction* mechanism.

The remainder of this paper is organized as follows. Section 2 explains our problem in detail, and the time-space network to mathematically formulate our problem. Section 3 presents an integer programme model by using path variables. And then we suggest a column-generation approach to find promising paths in the time-space network to improve the master problem at each iteration. In Section 4, we show the experimental results applied to the Korean high-speed railway network.

2 PROBLEM DEFINITION

Each operating company submits their favourite schedule of trains together with the routing plan. The routing plan means the sequence of the trains covered by each train set, so we can assume each train has its previous and next connection trains. And the Korean high-speed trains, so-called KTX, are being run with various halting patterns.

Our model can be easily explained on a time-space network where each node represents discretized time and each arc train movements. The approaches to use time-space network are found in other works [5–7].

In Fig. 1, each node on a horizontal line represents a discretized time while all the nodes together on any line belong to a single station. An arc between two nodes belonging to different stations represents the travelling times between those two stations. Where nodes corresponding to an intermediate station are split, the arc between them represents the dwell time. Each train can be expressed as a sequence of arcs. We assume each train has its own profit but the profit is decreased, depending on the deviation from the departure time at the original station and from requested dwell time at each stopping station.

And the trains are combined with each other by the train-set routing plan. For example, train 1 should be connected to train 2 and train 2 connected to train 4. If the departure time of train 1 would be delayed and there could not be enough turnaround time between the arrival time of train 1 and the departure time of train 2, the shift of departure time of train 1 should not be admitted. In this case, we say there are two routings, each of which has two trains. In the next section, we suggest an optimization model to consider the routing plan.

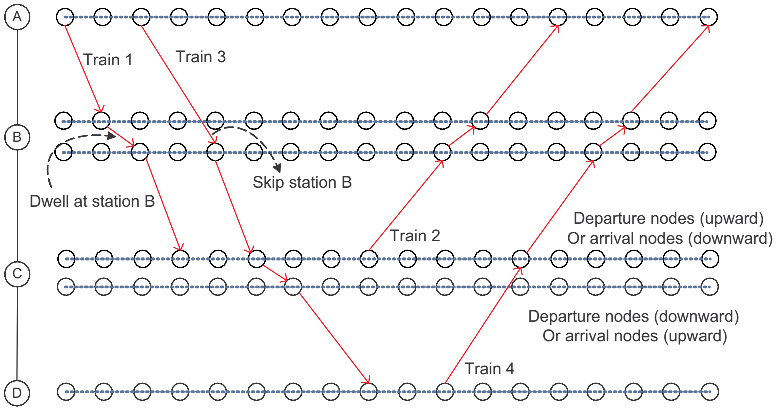

Figure 1: Representing the trains in the time-space network.

3 MODEL AND ALGORITHM

3.1 Optimization model

To develop our optimization model, the time-space network should be modified as follows. For the first train in a routing, we include an artificial source node, which is denoted as s_k, to express the time windows of departure at the original station. The size of the window is assumed to be a priori given by the infra manager.

And for the last train in a routing, we include an artificial terminal node, which is denoted as t_k for the routing k, and the associated arcs to express an alternative routing as a path to the node. As a result, a path from s_k to t_k corresponds to an alternative routing, which should be optimally adjusted in the model.

Finally, the arcs representing the dwelling time are added within a certain acceptable deviation. Figure 2 shows the time-space network representation of our model.

Operational constraints in the timetabling problem can be easily stated as *set packing* constraints [1], which means there is a restriction on the number of arcs in a set defining a *conflict*.

These constraints can be categorized as one of two types: those for guaranteeing the minimum headway at departure and arrival, and those for restricting the number of trains stopping simultaneously at any station. In fact we ignored the constraints preventing the trains from overtaking each other because we consider the case of the homogeneous traffic; in other words, all trains considered have the same running times between two stations. Figure 3 shows the way to express the operational constraints.

Based on the network representation as the above, we use the following notations to define our mathematical model.

Notation

- $G = (V, A)$: the time-space network. The nodes are indexed by i and j, and the arcs are a.
- c_a: the penalized profit of the arc a.

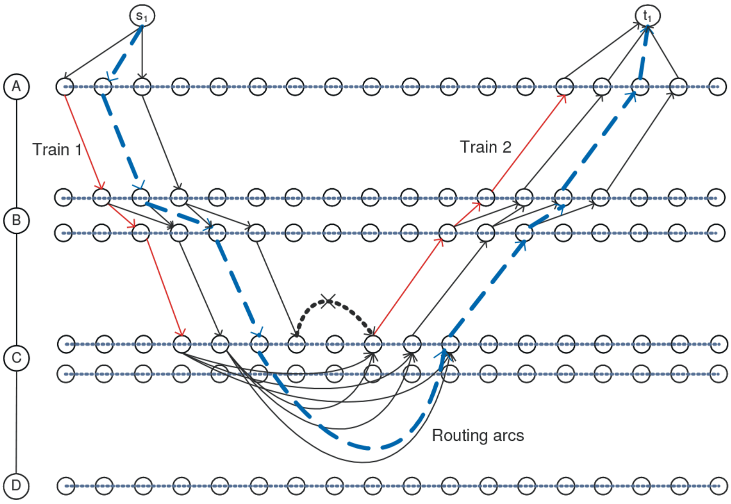

Figure 2: Track allocation considering train-set routing in the time-space network.

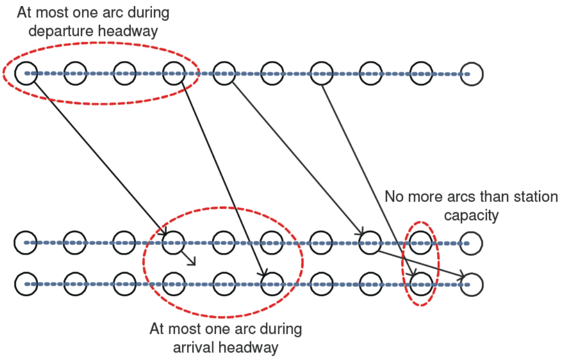
At most one arc during departure headway
No more arcs than station capacity
At most one arc during arrival headway

Figure 3: Establishing the conflict-free conditions in the time-space network.

- $A = A_1 \cup A_2 \cup A_3 \cup A_4 \cup A_5$ where $A_1, A_2 \ldots,$ and A_5 is the set of arcs which represent the train movement between two stations, dwelling at a station, connection to the next train in the routing plan, the arcs from the artificial source nodes and the arcs to the artificial terminal nodes, respectively.
- K: the set of request routing, the elements of which are indexed by k.
- T_k: the set of request trains belonging to the routing k, the elements of which are indexed by t.
- $P^k(t)$: the profit of the train t belonging to the routing k.
- P_k: the set of paths in G from S_k to t_k for the routing k, the elements of which are denoted by P.
- $c^k(P) = \sum_{a \in p} c_a$: the profit of the path P for the routing k. The path will be called *routing path*.
- C: the set of arcs that make conflicts. The set of C denoted by Ω.
- lc: the allowed number of arcs in the conflict set C.

We assume for simplicity that penalties are assigned only to the delayed departure time of the first train belonging to a routing and the delayed dwelling time at stations. If we assign the penalties to the associated arcs, we can easily compute the penalized profit of a request train. For the train connected from the previous train in a routing like train 2 in Fig. 2, the penalties are assigned to the *routing arcs* which means ones connecting two adjacent trains belonging to the same routing with sufficient turnaround time. In conclusion, we can compute the penalized profit of the routing path P by $\sum_{a \in p} c_a$.

In our study, we present a mathematical formulation based on path flow variables and a column-generation approach. The formulation, which is called *master problem* in the context of Dantzig-Wolfe decomposition [9], then becomes the following. If the path P is used for routing k, the decision variable $f^k(P)$ equals to 1, and $f^k(P)$ equals to 0 otherwise.

Master problem

$$\max \sum_{k \in K} \sum_{P \in P_k} c^k(P) f^k(P), \tag{1}$$

$$\text{s.t.} \sum_{P \in P_k} f^k(P) \leq 1, \forall k \in K, \tag{2}$$

$$\sum_{k\in K}\ \sum_{P\in P_k|P\cap C\neq\phi} f^k(P)\le lc, \forall C\in\Omega, \tag{3}$$

$$f^k(P)\in\{0,1\}, \tag{4}$$

where $c^k(P) = \sum_{a\in p\,(\in pk)} {}^{c}{}_{a}$.

The model maximizes the total penalized profit of all the routing paths according to eqn (1). By the eqn (2), the number of routing paths to be chosen is no greater than 1 for each routing. Finally, eqn (3) guarantees a conflict-free timetable.

3.2 Column-generation approach

If $f^k(P)\in\{0,1\}$ is replaced with $f^k(P)\geq 0$ and the non-negative dual variables are denoted as π_k, μ_C corresponding to eqn (2) and (3), respectively, the optimality condition of the relaxed master problem, which can be stated as dual feasibility condition, is as follows. For all $P\in P_k$,

$$\bar{c}^k(P) \quad = c^k(P) - \sum_{\{C|P\cap C\neq\phi\}} l_C\mu_C = \sum_{a\in P} ca - \sum_{a\in P}\ \sum_{\{C|a\in C\}} l_C\mu_C - \pi_k$$

$$\sum_{a\in P}[c_a - \sum_{\{C|a\in C\}} l_C\mu_C] - \pi_k \le 0. \tag{5}$$

We can find new columns to potentially improve the master problem by finding new paths $P\in P_k$ such that $c^{-k}(P) > 0$. To find such a path, we impose a modified cost on the arcs in the associated network with the routing k. It is then sufficient to find a longest path in the network, which can be easily solved because the network is acyclic [8]. Figure 4 shows the overall algorithm.

First, the initial relaxed master problem is established using the request trains and routing paths submitted by the operating companies. And then the above optimality conditions in eqn. 5 are checked, and if there is at least a routing path violating the optimality conditions,

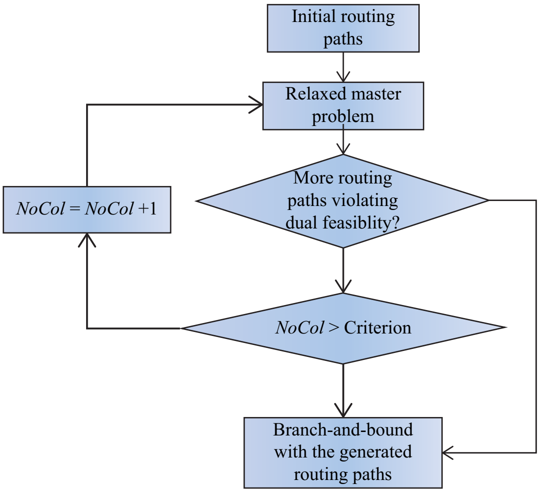
Initial routing paths
Relaxed master problem
More routing paths violating dual feasiblity?
NoCol = NoCol +1
NoCol > Criterion
Branch-and-bound with the generated routing paths

Figure 4: Column-generation heuristic.

new columns are added until the number of added columns, which is denoted as *NoCol* in Fig. 4, is no greater than 'Criterion'. If no more routing paths are found or *NoCol* is greater than 'Criterion', the algorithm goes in the branch-and-bound algorithm with the generated routing paths so far.

4 EXPERIMENTAL RESULTS

Figure 5 shows the Korean high-speed railway network used in our experiments. It consists of 40 stations and the bold lines (including dashed lines) indicate the dedicated tracks for the high-speed trains. The dashed lines show the dedicated tracks used by a new operating company SR. The incumbent operating company is running their high-speed trains (KTX) on the routes Seoul-Mokpo, Seoul-Yeosoo, Seoul-Jinjoo, Seoul-Busan and Seoul-Pohang. But the new entrant SR is supposed to run their high-speed trains (SRT) on the routes Suseo-Mokpo and Suseo-Busan.

For the experiments, we used the KORAIL's timetable in 2013 which consists of 200 trains a day. And to make the timetable of trains departing Suseo station, which means the new operating company's timetable, KORAIL's timetable was copied and then shifted to some minutes keeping the running times between new stations in minds. Totally, we generated 392 trains. And we assume that the driving time between two stations is fixed for both KTX and SRT. Therefore there is no need to include the conflict-sets preventing overtaking in Ω.

And we assume that there are 140 routings each of which contains one to six trains submitted by the companies. We simply generated 140 routings by keeping connecting trains if possible. So there are many routings with just one train. The following table (Table 1) shows the information about our routings.

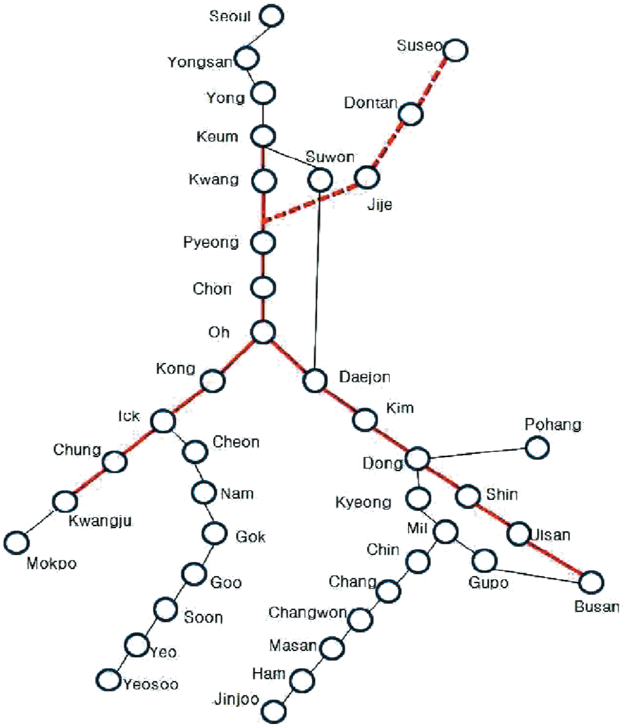

Figure 5: The Korean high-speed railway network.

Table 1: Routing information.

# of trains in routing	1	2	3	4	5	6
# of routings	52	13	19	30	19	2

For the dwell times, tolerances [0,5] are imposed on the dwell time so that the network includes five additional dwelling arcs for each request dwelling arc. And for departure times of the first train belonging to a routing, we set the tolerance interval as [0,10] . As mentioned before, the profit is penalized, depending on the deviation from the request times. We set the penalty 1 for each minute delay. The profits of trains are all set to be a constant. Finally, the algorithm terminates if 'Criterion' in Fig. 4 becomes no smaller than 7,000. The algorithm was implemented using CPLEX 12.5 on a PC with an Intel Core 3.20 GHz CPU.

We implemented our model by two steps: the first step is to construct the time-space network and the set of conflict sets Ω given the list of trains and routing plans by the operating companies. In case of Fig. 5, it took about 19 min to construct the overall time-space network and Ω. The second step is to find an optimal routing paths by Fig. 4. Table 2 shows the performances of our algorithm in Fig. 4.

As shown in the table, it was assumed that two operating companies submitted 392 trains and 140 routings, and all the trains have the profit of 100.

First, we investigated the difference between the solutions of two cases $d = 0$ and $d = 10$. $d = 0$ means there is no room to adjust the request times of trains so that the number of chosen trains as well as routings are much less than the case of $d = 10$.

Table 3 shows how many trains or routings were adjusted and how much time was shifted. The first row indicates the total amount of shifted times that resulted in our algorithm and the second and third rows show the number of routings and trains with no more than the associated delay, respectively.

However, it takes a lot of time to find a solution even though the number of generated columns is restricted to 7,000. In fact, most of computation times was taken to find a good integer solution by branch-and-bound procedure even though the MIP gap was set to be 15% in CPLEX.

Table 2: Experimental results.

Departure [0, d]	# of routings: $\|K\|$	# of trains: $\sum_{k\in K}\|T_k\|$	Total profit	Computation time (min.)
given	140	392	39,200	n/a
$d = 0$	101	259	25,543*	4
$d = 0$	108	285	27,640*	31

* terminated if MIP gap is within 15%.

Table 3: Total delay.

Delay (min.)	= 0	≤ 5	≤ 10	≤ 20	≤ 60
# of routings	33	63	82	94	108
# of trains	124	234	261	285	285

5 CONCLUDING REMARKS

We suggested a new optimization model and algorithm for the track allocation model to consider the train-set routing plan. The formulation was based on path-flow variable defined on the time-space network, and then a column-generation approach was presented. But the algorithm should be more elaborated since there still needs a lot of time to find an integer solution by branch-and-bound procedure.

ACKNOWLEDGEMENT

This article was supported by the Korea National University of Transportation in 2016.

REFERENCES

[1] Borndörfer, R., Grotschel, M., Lukac, S., Mitusch, K., Schlechte, T., Schultz, S. & Tanner, A., An auctioning approach to railway slot allocation. *ZIB-Report*, 05-45, 2005.

[2] J.-E. Nilsson., Allocation of track capacity: Experimental evidence on the use of priority auctioning in the railway industry. *International Journal of Industrial Organization*, **17**, pp. 1139–1162, 1999 DOI: 10.1016/S0167-7187(99)00016-8.

[3] Brewer, P.J. & Plott, C.R., A binary conflict ascending price (bicap) mechanism for the decentralized allocation of the right to use railroads tracks. *International Journal of Industrial Organization*, **14**, pp. 857–886, 1996 DOI: 10.1016/0167-7187(96)01014-4.

[4] Parkes, D.C. & Ungar, L.H., An auction-based method for decentralized train scheduling. *Proceedings of the Fifth International Conference on Autonomous Agents*, 2001.

[5] Caprara, A., Fischetti, M. & Toth, P., Modeling and solving the train timetabling problem. *Operations Research*, **50**(**5**), pp. 851–861, 2002.

[6] Caprara, A., Cacchiani, V. & Toth, P., A column generation approach to train timetabling on a corridor. *4OR*, **6**, pp. 125–142, 2008.

[7] Caprara, A., Cacchiani, V. & Toth, P., Scheduling extra freight trains on railway networks. *Transportation Research Part B*, **44**, pp. 215–231, 2010 DOI: 10.1016/j.trb.2009.07.007.

[8] Ahuja, R.K., Magnanti, T.L. & Orlin, J.B., *Network Flows : Theory, Algorithms, and Applications*. Prentice Hall: Upper Saddle River, NJ, 1993.

[9] Wosley, L.A., *Integer Programming*. John Wiley and Sons: Hoboken, NJ, 1998.