ELSEVIER

A pragmatic algorithm for the train-set routing: The case of Korea high-speed railway ☆

Sung-Pil Hong[a], Kyung Min Kim[b], Kyungsik Lee[c, c], Bum Hwan Park[d, *]

[a]*Department of Industrial Engineering, Seoul National University, Republic of Korea*
[b]*Railway Policy and Logistics Department, Korea Railroad Research Institute, Republic of Korea*
[c]*School of Industrial and Management Engineering, Hankuk University of Foreign Studies, Republic of Korea*
[d]*Department of Management Information System, Korea National Railroad College, 374-18, Worm-Dong, Uiwang-Si, Gyeonggi-Do, 437-763, Republic of Korea*

Abstract

This paper presents a two-phased train-set routing algorithm to cover a weekly train timetable with minimal working days of a minimal number of train-sets. First, relax maintenance requirements and obtain minimum cost routes by solving the polynomial relaxation. Then, maintenance-feasible routes are generated from the crossovers of the minimum cost routes. This pragmatic approach seems particularly effective for the high-speed railway systems, where the railway topology is relatively simple with few end stations while the trains are frequent. Applied to the current weekly timetable of the Korea high-speed railway, we could find an optimal feasible routing, which is an 8.8% improvement over the current routing generated by a set partitioning approach based on a path generation scheme.
© 2008 Elsevier Ltd. All rights reserved.

Keywords: Rail transport; Train-set routing; Minimum cost flow; Maintenance

1. Introduction

In April, 2004, after more than a decade of preparatory project, the KORAIL (Korea Railroads, Corp.) initiated the historical KTX (Korea Train eXpress) service. The KTX is a high-speed railway system that operates trains at the speed of 300 km per hour. One of the important issues of the KTX is to cover the timetable with

a minimum rolling stock. The KTX is expensive to operate. After the first three-month operation, KORAIL estimated the average annual operation cost per single train-set to be more than 16 billion won (about 16 million US dollars). More importantly, by doing so, in its early stage of service, KORAIL may have the flexibility to adapt to the evolving demand of the KTX.

The current solution is generated from a set partition formulation solved by the column generation method, a typical integer programming approach for the fleet routing with maintenance requirements which makes the problem intractable, as we will see later. To achieve an improvement over the current implementation, we need to fully exploit the key characteristics of a high-speed railway. They operate on a railway, typically

simpler than conventional one, with few end stations. But, the trains are frequent (so that size of the problem still makes an exact solution of the integer program impractical). Hence, the route of a train-set is more likely to cross with the route of another at the same stations within the time intervals that are short enough. Then, the subroutes of the two train-sets after the station can be interchanged to yield a new routing employing the same level of rolling stock. This implies that there are many alternative solutions with the same objective value. This observation motivates us to consider a pragmatic approach. First, relax the maintenance requirements and obtain minimum cost routes by identifying polynomial structure of the relaxed problem. Then, generate systematically the crossovers of the routes of train-sets to obtain a solution satisfying the maintenance requirements.

As a result, we could get an 8.8% improvement in the number of total train-set days over the current implementation. In particular, we obtained an optimal solution for the current weekly timetable. Our method seems promising: for a set of simulated timetables with additional trains, it provided optimal solutions in most cases. This paper summarizes the model, the solution approach, and the experimental results. In Section 2, we classify the train-set routing problem and summarize the previous studies. Section 3 is devoted to defining the KTX problem and the NP-hardness proof of the specific problem. In Sections 4 and 5, respectively, we discuss the heuristic algorithm and its experimental results applied to the KTX weekly schedule.

2. Train-set routing problem

Here are some terminologies this paper adopts (see, e.g. [1]). By a *train*, we mean a particular service specified by two end stations along with intermediate stations, the departure and arrival times of each station, and the type of a train-set assigned to each train. The physical unit of rolling stock to cover a train is called a *train-set*. Hence, a train-set is a set of passenger cars and power unit(s).

Our goal is to cover the timetable with a minimal level rolling stock as will be specified in detail later. This problem is known as the *rolling stock rostering problem* [2] or *vehicle scheduling problem* [3] which, in its general sense, consists of the *capacity allocation problem* [1] or the *train length problem* [2] that decides the capacity of trains in the schedule, and the *train-set routing* [1] or the *train assignment* [2] that assigns the rolling stock to the scheduled trains.

The rostering problem dealing with heterogeneous rolling stocks has been considered in the various stud-

ies that can be conveniently categorized as the *locomotive assignment problem* [4,5], the *car assignment problem* [6], or the simultaneous locomotive and car assignment problem [7]. The mathematical formulation is typically based on the integer multi-commodity flow problem whose commodities correspond to the rolling stock types that can be assigned to each train [4,7].

The KTX is, however, currently operated with homogeneous rolling stock: there is a single train-set type composed of 18 passenger and two power cars. The homogeneity makes the train length problem unnecessary and simplifies our problem into the train-set routing problem.

A train-set routing is determined by the number of train-sets and the route, namely the sequence of trains, that each train-set traverses in a time horizon. A *train network* is illustrated in Fig. 1. As in the figure, each horizontal line indicates the time axis at each end station, A, B, and C, aligned vertically. (Here, A', for instance, denotes the same end station as A but at the end of the time horizon.) Hence each train can be represented by an arc whose end points on the horizontal lines are uniquely determined by its departure and arrival time at the end stations. Therefore, the train-set routing problem is equivalent to covering all the train arcs with the train-arc disjoint paths. This problem can be classified into three categories:

Cover-by-paths: This problem simply requires each arc to be covered by a path. Hence, in particular, there may be an end station where the number of incoming train-sets at the end of a time horizon is different from the number of out-going train-sets at the beginning of the next time horizon. Then it entails "deadhead" or empty trains.

Cover-by-circulation: If the routing does not allow a deadhead train in the above sense, we will call it a *circulation*. It is not hard to see that if trains are symmetric in both directions between the end stations, a circulation exists.

Cover-by-rotation: By a *rotation*, we mean a circulation in which, as the time horizon repeats, a single train-set covers all the trains. This means every train-set, in the long run, covers the same set of routes. A rotation is a desirable practice in that it maintains train-sets and rails in a homogeneous condition. To decide if a train network has a rotation is easy. This is equivalent to finding an Eulerian walk in the graph obtained from the train network by shrinking the lines corresponding to each station (e.g. lines $A - A'$, $B - B'$, and $C - C'$ in Fig. 1) into single nodes. If the graph is Eulerian, as usually the case, then it has a rotation. However, if connecting trains incurs costs or revenues which depend
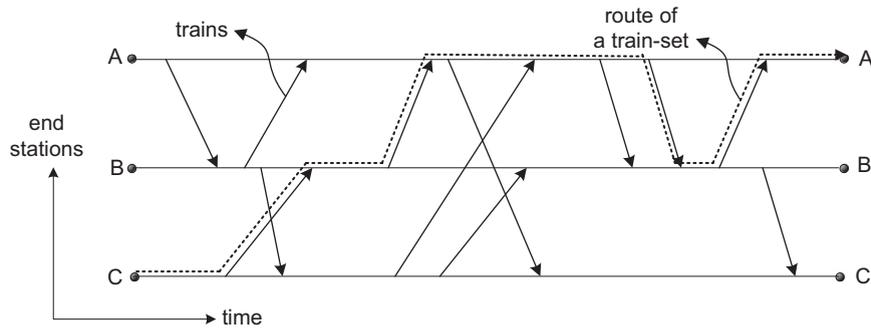
Fig. 1. A train network.

Table 1
Previous studies

Authors	Time	Dist.	Sol. approach	Category
Khedher et al. [1]	○	○	Set partit.+path generation	Path
Chung et al. [9]	×	×	MIP formul.+genetic algo.	Path
Gopalan and Talluri [10]	○	×	Euler Tour+fix LOF.	Rotation
Clarke et al. [8]	○	×	TSP heuri. Lagr. relax.	Rotation
Zhao et al. [11]	○	×	TSP heuri.+Random choice	Rotation
Kabbani and Patty [12]	○	×	Set partit.+path generation	Circul.
Schrijver [13]	×	×	Min-cost-flow	Circul.
Erlebach et al. [14]	○	×	Bipart. matching+extra trains for mainten.	Circul.
Anderegg et al. [2]	○	○	Min-cost-flow+ extra trains for mainten.	Circul.
Our paper	○	○	Min-cost-flow+crossover among cycles.	Circul.

on the connected pair of trains, then the order of the arcs in the walk matters. Then the problem of finding the best Eulerian walk is reduced to a TSP in the corresponding line graph, which is NP-hard. For example, Clarke et al. [8] considered such a model by introducing *through value*, a revenue expected when a desirable pair of flights are connected by the same aircraft.

From the intensive literature on the train-set or aircraft routing problem, Table 1 summarizes the studies, like ours, dealing with homogeneous rolling stock or aircraft according to the above categories. The second and third column, respectively, indicated by ○ if the model considers the maintenance requirements based on the time or distance driven by a train-set, and by ×, otherwise. Among them, the models of [2,14] appear similar to ours in the sense that they pursue a cover-by-circulation meeting the maintenance requirements in terms of both time and distance even though the requirements are not so specific or restrictive as the KTX case. They proposed a solution method which first finds a routing with maintenance requirements relaxed based on the bipartite matching [14] or the min-cost flow [2] formulation. Then, a heuristic is employed to incrementally increase the number of train-sets to meet the maintenance requirements.

Our study is different from the previous ones including [2,14] in two things. First, we consider a hierarchical multi-objective. To do so, the usual train network is elaborated to reflect the second objective. Second, we have very specific maintenance requirements in both time and distance. They are among the most restrictive types which a train-set routing problem may have in the literature. We will also provide a proof that the particular maintenance requirement only in time or distance makes the problem NP-hard. We also propose a heuristic to obtain a maintenance routing based on the optimal solution from the relaxation, which fully exploits the characteristics of a high-speed train network discussed later. The heuristic is so effective for the KTX case that it produce, in a few seconds, optimal routings not only for the current KTX schedule but also for a set of schedules extended according to the practice of the KORAIL.

3. The KTX problem

As mentioned earlier, the KTX rolling stock is homogeneous: it uses a single type of train-set. Currently, 46 sets are available. On the claw-like rail network, there are 19 KTX stations and seven of them with names are end stations. (See Fig. 2.) As on January 2006,

Fig. 2. KTX rail network and stations.

Table 2
Sample trains from the Tuesday timetable

Train	Origin	Depart. Time	Destin.	Arriv. Time	Driv. Dist
3	Seoul	AM 7:00:00	Busan	AM 9:40:00	409
113	Seoul	PM 1:45:00	Daegu	PM 3:32:00	293
23	Seoul	PM 5:00:00	Busan	PM 7:37:00	409
48	Busan	AM 8:30:00	Seoul	AM 11:27:00	409
60	Busan	PM 2:30:00	Seoul	PM 5:10:00	409
205	Yongsan	AM 8:25:00	Mokpo	AM 11:50:00	354
233	Yongsan	PM 4:40:00	Gwangju	PM 7:23:00	411
154	Daejeon	AM 7:10:00	Seoul	AM 8:11:00	160
120	Daegu	PM 4:15:00	Seoul	PM 6:02:00	293
240	Gwangju	AM 10:00:00	Yongsan	PM 12:48:00	411
210	Mokpo	AM 10:40:00	Yongsan	PM 1:53:00	354

KORAIL runs 972 KTX trains a week. The numbers of daily trains are (Mon, Tue, Wed, Thr, Fri, Sat, Sun) = (135, 132, 132, 132, 137, 152, 152). See Table 2 for some sample trains from the current Tuesday timetable.

3.1. The objective

The KORAIL has two objectives with different priorities in the KTX train-set routing. First, they want to cover the weekly timetable with the minimum number of train-sets. Second, among such solutions, they want to find one that minimizes the total working days of train-sets, or the total *train-set days* over a schedule cycle is minimized. More specifically, the total train-set days of a routing over a schedule cycle is the sum of the numbers of train-sets operated daily over the seven days of a week. If a train-set can be reserved from an operation for a whole day, it not only saves operating and indirect costs but also can be used in an emergency. Moreover, it is also a practical way to comply with the long-term checks which are not normally reflected in daily or weekly schedule.

Such a routing can be obtained, as will be discussed in Section 4.1, by solving the minimum cost circulation problem on the train network in Fig. 1 modified with some additional types of arcs and their costs.

3.2. The constraints

Maintenance: Each KTX train-set should be given maintenance if one of the time elapsed and the distance driven, since the last maintenance, has reached a predetermined value. There are six levels of maintenance from *daily checks* (3 days or 3500 km) to *overall checks* (1 year or 800,000 km). In our model, only daily checks are considered as they are the only checks whose time or distance interval is smaller than our time horizon, a week. Daily checks are performed overnight only at Seoul and Busan Station.

Turn-around times: By a turn-around time, we mean a minimum time required for a train-set to depart for the next train after it arrives at an end station completing a train. Turn-around time depends upon stations.

Deadhead trains: Although deadhead trains may improve the utilization of rolling stock in the presence of geographical imbalances of rolling stock availability [15], they are not allowed as a policy in the KTX and are restricted to the exceptional cases.

3.3. NP-hardness of maintenance routing

A vehicle routing problem with additional maintenance constraint(s) is typically NP-hard. For instance, Erlebach et al. [14] established the NP-hardness and some inapproximability results of the train-set routing problem with the maintenance requirement that every train-set needs to pass through the maintenance station before it completes its routing cycle (irrespective of the time or distance it takes to complete the cycle). To the authors' best knowledge, however, none of the previous vehicle routing problems is identical to our KTX train-set routing problem. Hence, we will provide a simple NP-hardness proof of the KTX train-set routing
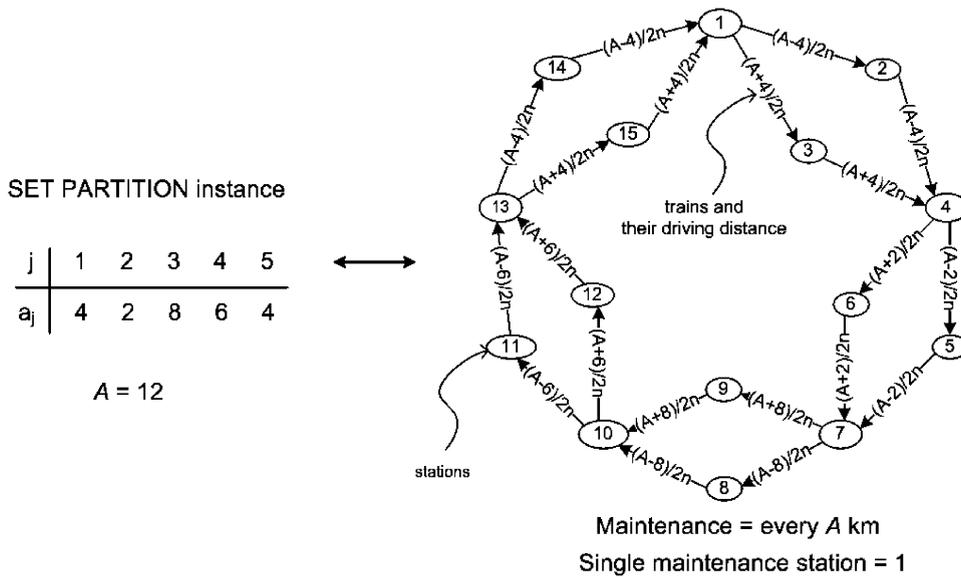
Fig. 3. SET PARTITION is yes iff there is a feasible routing using two train-sets.

problem. It is intractable even when there is a single (time or distance) maintenance constraint.

Consider the NP-complete problem, SET-PARTITION which is the problem of finding a partition S, $N \backslash S$ of a given set of positive integers a_j, $j \in N := \{1, 2, \ldots, n\}$ so that the sums of numbers in both sets are the same, namely $\sum_{j \in S} a_j = \sum_{j \in N \backslash S} a_j = A$, (hence $\sum_{j=1}^{n} a_j = 2A$). We will show that SET-PARTITION is polynomially reducible to the decision problem asking if a KTX train-set routing instance has a routing with the number of used train-sets less than or equal to some predetermined value.

The corresponding train network has $3n$ end stations and $4n$ trains. Maintenance checks should be done in every A km at node 1, the only maintenance station. For each a_j, we consider four trains represented by the arcs $(3j - 2, 3j - 1)$, $(3j - 1, 3j + 1)$ with travelling distance $(A - a_j)/2n$, and $(3j - 2, 3j)$, $(3j, 3j + 1)$ with travelling distance $(A + a_j)/2n$, where the node $3n + 1$ is identified with node 1.

Fig. 3 illustrates an example of the reduction for SET-PARTITION instance given by $a = (4, 2, 8, 6, 4)$ and hence $A = 12$.

The claim is the answer to SET-PARTITION is "yes" if and only if we can cover all the trains with two train-sets. But, notice that the sum of the driving distances of trains is $2A$, and hence the driving distances of two train-set are $A + \alpha$ and $A - \alpha$ for some $\alpha > 0$. Hence, all the trains can be covered by two train-sets meeting the A km maintenance requirements if and only if the

circles traversed by two train-sets have the same driving distance A. Choose any of the two circles. Define a set S as follows: $j \in S$ if and only if the two consecutive trains with the driving distance $(A + a_j)/2n$ are on the circle. Then, it is easy to check that the fact that the circle distance is A implies $\sum_{j \in S} a_j = \sum_{j \in N \backslash S} a_j = A$, and vice versa. Hence the validity of reduction follows.

4. The algorithm

4.1. Min-cost routing without maintenance

Without the maintenance constraints, the KTX train-set routing problem is solvable in polynomial time. The problem can be formulated as a min-cost circulation problem [13,16,17] or as a matching problem [14,18]. Such a formulation dates back as early as 1950s [16], and since then has recurrently and independently appeared in the literature (see, e.g. [13,14,17,18]).

In the KTX case, however, we need a formulation properly reflecting the two objectives discussed in Section 3.1 according to their priorities. To do so, we need to modify the train network of Fig. 1 to encourage train-sets to stay as many days as possible while minimizing the number of train-sets utilized during a week.

Fig. 4 illustrates the modified network corresponding to the specific train network from Fig. 1. It is composed of six types of arcs, i.e. the *train arcs*, the *train-connect arcs*, the *aggregation arcs*, the *disaggregation arcs*, the *one-day-stay arcs*, and the *backward*
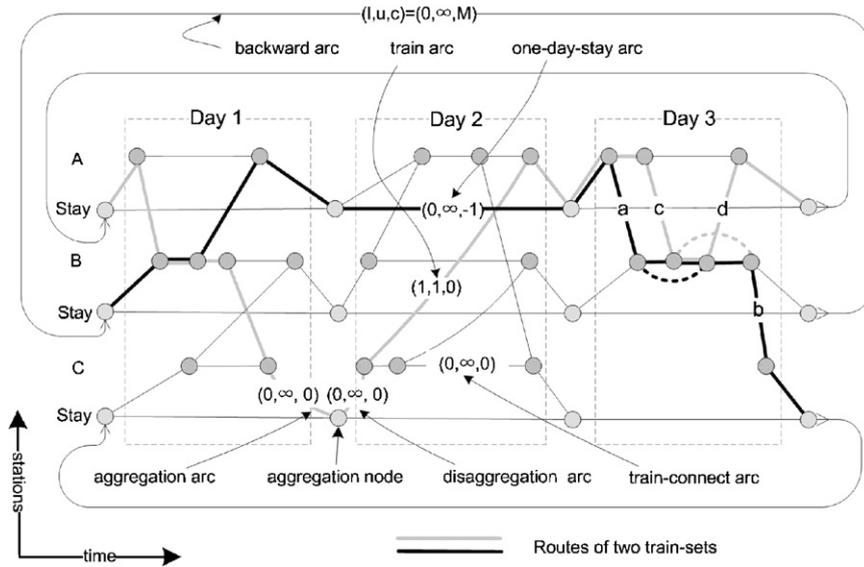
Fig. 4. The modified train network and two examples of train-set route.

arcs. The train arcs, which are the same as the train arcs from the original train network of Fig. 1, represent the trains to be covered by a train-set. And it is enforced by setting both lower and upper bounds equal to 1. But, every train arc is assigned the cost of 0. The end node of a train arc represents the arrival time plus the required turn-around time.

A train-connect arc represents the waiting of a train-set at the same station until the next train. These arcs are assigned a lower bound, an upper bound, and costs $(l, u, c) = (0, \infty, 0)$.

The aggregation nodes, as positioned on the date lines, aggregate the train-sets that stayed overnight at the end stations. The arcs incoming to aggregation node are the aggregation arcs and those outgoing from aggregation node are the disaggregation arcs which are all assigned $(l, u, c) = (0, \infty, 0)$.

Similarly, the backward arcs, connecting the end and beginning time epochs of two consecutive time horizons (weeks, in the KTX case) of the same end stations, collectively aggregate all the train-sets utilized during a week. Therefore, by assigning parameters $(l, u, c) = (0, \infty, M)$ for a sufficiently large M, we can minimize the total number of the train-sets to cover the weekly schedule, the first objective of the KTX train-set routing. Finally, the one-day-stay arcs are to accommodate the train-sets that stay waiting a whole day at the stations. These arcs are assigned $(l, u, c) = (0, \infty, -1)$ so that a train-set staying a day incurs a unit revenue. By doing so, we encourage the train-sets to stay a day from daily

operation and hence minimize the total train-set days of a routing.

From the cost parameters assigned to the backward and one-day-stay arcs, a min-cost circulation will provide a train-set routing that minimizes the total train-set days among the solutions with the smallest possible number of train-sets that can cover the weekly schedule.

4.2. Crossover heuristic for maintenance routing

A train-set route is feasible if and only if it stays overnight at Seoul or Busan Station before the train-set has run 3500 km or 3 days have elapsed since the last maintenance. There is no guarantee that a solution of the min-cost circulation formulation satisfies this condition. However, due to the characteristics mentioned earlier, there tend to be numerous alternative solutions. This motivates us to consider a maintenance routing heuristic based on the generation of crossovers of the minimum cost routes obtained from the min-cost circulation formulation.

In [19], it has been shown that the problem of finding a routing meeting the maintenance requirements by crossing over is itself NP-hard. They also used an integer programming formulation of the problem. In the KTX case, however, the following simple heuristic turns out to be very effective.

It is easy to see from Fig. 4, a feasible circulation uniquely determines the daily subroutes of a train-set. A (weekly) route is the concatenation of the daily
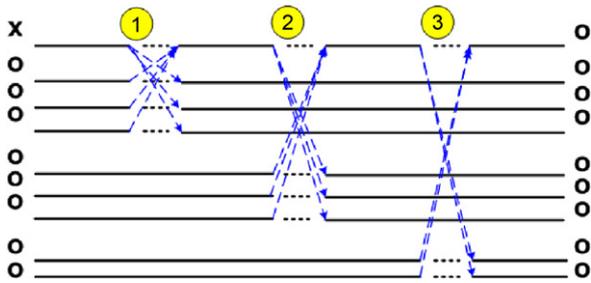
Fig. 5. Crossover generation.

subroutes. Each daily subroute is one of the four types: a subroute starting and ending at maintenance stations $((\circ, \circ)$-type), starting at a maintenance station but ending at a non-maintenance station $((\circ, \times)$-type), starting at a non-maintenance station but ending at a maintenance station $((\times, \circ)$-type), and finally starting and ending at non-maintenance stations $((\times, \times)$-type).

Let R_1 and R_2 be two disjoint routes. If R_1 and R_2, respectively, contain two consecutive trains, (i_1, j_1) and (i_2, j_2) satisfying following conditions, then we can generate two crossovers: (1) the trains i_1 and i_2 have the same destination, and (2) the arrival time of i_1+ turn-around time $\leqslant$ departure time of j_2 and arrival time of i_2+ turn-around time $\leqslant$ departure time of j_1. For example, from the two routes of thick edges in Fig. 4, two crossovers can be generated as the above conditions are satisfied among the trains, $i_1 = a$, $j_1 = b$, $i_2 = c$, and $j_2 = d$ at Busan Station on Day 3.

The proposed heuristic has two pragmatic ideas. First, obtain $(\circ, \times)$- and $(\times, \circ)$-type daily subroutes by generating crossovers from the pairs of $(\circ, \circ)$- and $(\times, \times)$-type daily subroutes. This is motivated by the observation that as the train-sets are highly utilized, the distances driven get close to the limit, 3500 km, before 3 days. Hence, $(\times, \times)$-type daily subroutes are most likely the source of infeasible routes. Second, we try further to reduce the driving distance of an $(\circ, \times)$- and $(\times, \circ)$-type daily subroutes by generating crossovers with $(\circ, \circ)$-type daily subroutes. For each $(\circ, \times)$- or $(\times, \circ)$-type daily subroute, this is repeated over $(\circ, \circ)$-type daily subroutes from the same day and station. (See Fig. 5.) This is to prevent a two-day subroute that is the concatenation of an $(\circ, \times)$- and a $(\times, \circ)$-type daily subroutes in the order, whose driving distance exceeds 3500 km. Such subroutes tend to appear more frequently as the train-sets get fully utilized.

5. The experimental results

Table 3 shows the solutions obtained by using the current KTX timetable. Comparing the numbers from the last two rows, we can see that our solution is an exact optimum as its objective values coincide with the optimal value of min-cost circulation problem before applying the crossover heuristic. As summarized in the table, the number of train-sets utilized for the weekly timetable reduced from 39 to 37. Also, on each day of a week, we can cover the trains with less train-sets; overall, the sum of train-sets over the seven days of a week, namely, the total number of train-set days reduced from 259 to 236, an 8.8% improvement.

Next, we test the crossover heuristic on an additional set of timetables. The KORAIL normally revises timetable for only small portion of trains. Since April 2004, the number of weekly KTX trains has increased from its initial 924 to the current 972. Following this practice, we generated 10 timetables by incrementally increasing the number of trains by two at each time proportionally to the numbers of trains with the same end stations. Timetable $j + 1$ is obtained from Timetable j as follows:

Timetable 1	Add 1 Seoul–Busan and 1 Busan–Seoul train on every day
Timetable 2	Add 1 Seoul–Busan and 1 Busan–Seoul train on every day
Timetable 3	Add 1 Seoul–Busan and 1 Busan–Seoul train on every day
Timetable 4	Add 1 Yongsan–Mokpo and 1 Mokpo–Yongsan train on every day
Timetable 5	Add 1 Seoul–Daegu and 1 Daegu–Seoul train on every day
Timetable 6	Add 1 Seoul–Busan and 1 Busan–Seoul train on every day
Timetable 7	Add 1 Yongsan–Mokpo and 1 Mokpo–Yongsan train on every day
Timetable 8	Add 1 Seoul–Busan and 1 Busan–Seoul train on every day
Timetable 9	Add 1 Seoul–Daegu and 1 Daegu–Seoul train on every day
Timetable 10	Add 1 Yongsan–Gwangju and 1 Gwangju–Yongsan train on every day

The results are summarized in Table 4. For all the timetables, we could obtain exact optimal solution. This confirms our intuition that the characteristics of the high-speed railway would make the crossover heuristic particularly effective.

The obtained routes are, of course, all circulations. In most cases, the solution is not a rotation after the minimum cost routing. Interestingly, however, after

Table 3
Optimal solutions for current timetable

	Total no. train-sets	Mon	Tue	Wed	Thr	Fri	Sat	Sun	Time[a]	Dist.[a]
Current	39	36	36	36	36	37	39	39	0	0
Min circulation	37	32	32	32	32	34	37	37	4	10
W. maint. requir.	37	32	32	32	32	34	37	37	0	0

[a] 'Time' and 'Dist.' columns indicate the number of routes violating maintenance requirements.

Table 4
Optimal solutions for additional timetables

	Mon	Tue	Wed	Thr	Fri	Sat	Sun	Time	Dist.	Max. dist. (km)	OPT.	CPU (s)
Table 1	34	34	34	34	36	39	39	0	0	$\leqslant 3500$	Y	2.62
Table 2	36	36	36	36	38	40	40	0	0	$\leqslant 3500$	Y	2.39
Table 3	36	36	36	36	38	41	41	0	0	$\leqslant 3500$	Y	1.97
Table 4	38	38	38	38	40	43	43	0	0	$\leqslant 3500$	Y	2.73
Table 5	39	39	39	39	41	45	45	0	0	$\leqslant 3500$	Y	2.19
Table 6	40	40	40	40	42	45	45	0	0	$\leqslant 3500$	Y	2.26
Table 7	41	41	41	41	43	46	46	0	0	$\leqslant 3500$	Y	3.12
Table 8	42	42	42	42	44	46	46	0	0	$\leqslant 3500$	Y	2.95
Table 9	42	42	42	42	44	45	45	0	0	$\leqslant 3500$	Y	3.23
Table 10	43	43	43	43	45	47	47	0	0	$\leqslant 3500$	Y	2.84

The numbers of train-sets utilized during a week coincided with those for the weekends.

application of the crossover heuristic, every route turned into a rotation for this particular set of timetables.

6. Further research

An immediate question is how far the crossover heuristic can be effective. As observed earlier, the heuristic seems effective if the problem has many alternative routes, which seems to be the case for most high-speed express rail systems. Thus, it will be an interesting study to apply the heuristic to other high-speed rail systems.

Acknowledgement

References

[1] Khedher NB, Kintanar J, Queille C, Stripling W. Schedule optimization at sncf: from conception to day of departure. Interfaces 1998;28:6–23.

[2] Anderegg L, Eidenbenz S, Gantenbein M, Stamm C, Taylor DS, Weber B, et al. Train routing algorithms: concepts, design choices, and practical considerations. In: Proceedings of the fifth workshop on algorithm engineering and experiments (ALENEX03); 2003.

[3] Bunte S, Kliewer N. An overview on vehicle scheduling models. DSOR Working Papers: University of Paderborn; 2006.

[4] Ziarati K, Soumis F, Desrosiers J, Gelinas S, Saintonge A. Locomotive assignment with heterogenous consist at CN North America. European Journal of Operational Research 1997;97:281–92.

[5] Forbes MA, Holt JN, Watts AM. Exact solution of locomotive scheduling problems. Journal of the Operational Research Society 1997;42(10).

[6] Lingaya N, Cordeau JF, Desaulniers G, Desrosiers J, Soumis F. Operational car assignment at VIA rail Canada. Transportation Research B 2002;36:755–78.

[7] Cordeau J-F, Desaulniers G, Lingaya N, Soumis F, Desrosiers J. Simultaneous locomotive and car assignment at VIA rail Canada. Transportation Research Part B—Methodological 2001;35(8):767–87.

[8] Clarke L, Johnson EL, Nemhauser GL. The aircraft rotation problem. Annals of Operations Research 1997;69(1):33–46.

[9] J.-W. Chung, S.-M. Oh, and I.-C. Choi, A hybrid genetic algorithm for train sequencing in the korean railway, OMEGA—International Journal of Management Science, in press, doi:10.1016/j.omega.2007.12.001.

[10] Gopalan R, Talluri KT. The aircraft maintenance routing problem. Operations Research 1998;46(2).

[11] Zhao P, Tomii N, Fukumura PN, Sakaguchi T. An algorithm for train-set scheduling based on probabilistic local search. In: Computers in Railways VIII; 2002. p. 817–26.

[12] Kabbani NM, Patty BW. Aircraft routing at american airlines. In: Proceedings of the 32nd annual symposium of AGIFORS, Budapest, Hungary; 1992.

[13] Schrijver A. Minimum circulation of railway stock. CWI-Quarterly 1993;6(3):205–17.

[14] Erlebach T, Gantenbein M, Hürlimann D, Neyer G, Pagourtzis A, Penna P, et al. On the complexity of train assignment problems. In: Proceedings of ISAAC01, lecture notes in computer science, vol. 2223, 2001. p. 390–402.

[15] Kuo C-C, Nicholls GM. A mathematical modeling approach to improve locomotive utilization at a freight railroad. OMEGA—International Journal of Management Science 2007;35:472–85.

[16] Dantzig G, Fulkerson D. Minimizing the number of tankers to meet a fixed schedule. Naval Research Logistics Quarterly 1954;1:217–22.

[17] Brucker P, Hurink J, Rolfes T. Routing of railway carriages. Journal of Global Optimization 2003;27:313–32.

[18] Bertossi AA, Carraresi P, Gallo G. On some matching problem arising in vehicle scheduling models. Networks 1987;17: 313–32.

[19] Maróti G, Kroon L. Maintenance routing for train units: the interchange model. Computers & Operations Research 2007;34(4):1121–40.