Application of Rule Mining Technology for Train-order Rescheduling

Takashi SAKAGUCHI
Transport Operation Systems Laboratory, Signalling and Transport Information Technology Division

Keisuke SATO
Transport Operation Systems Laboratory, Signalling and Transport Information Technology Division
(Currently Kanagawa University)

One of the technical problems encountered when dispatching trains using computation is how to reflect all the conditions or criteria that influence a decision that form part of the tacit knowledge that train dispatchers have acquired from experience in conventional rescheduling systems. A method has been developed from operation performance records, to generate decisions that would have been made by dispatchers with a high probability under specific conditions, namely operational dispatching rules (ODRs). A train rescheduling algorithm was created based on the mechanisms related to ODRs and its validity by comparing actual operation planning results with computed outputs.

Keywords: train rescheduling, operational dispatching rule, sequential rule mining, association rule, mathematical optimization

1. Introduction

When train schedules are disrupted because of an accident or bad weather, and when the situation cannot resolve itself, then operation plans have to be changed. Dispatchers must then decide how to reschedule trains based on basic policies of the operation center corresponding to the state of the traffic at the time.

Using ICT, RTRI has been conducting research to find solutions for this type of situation by using mathematical optimization to develop a method to automatically generate rescheduling proposals [1]. The developed method is designed to find the optimal rescheduling using pre-defined quantitative evaluation indices. However, it is difficult to incorporate the knowhow of experienced dispatchers in such a method, so the challenge was to improve the system to take such implicit knowledge into account. Consequently, this study conducted rule mining on past dispatcher decision data, to create a system that would automatically generate rescheduling proposals. This paper presents the results of this study.

2. The subject of the research

2.1 Train rescheduling

The approach used to reschedule operation plans depends on the scale of the disruption: in the case of small scale disruptions, local solutions can be applied to recover operations following a disruption, such as adjusting the headway of a group of trains (headway adjustment), changing train running order through stations and level crossings (change in running order), changing the platform used by trains at OD or intermediary stations (platform change), etc.

Major disruptions however, require a different approach, based on the size of the area affected: first, dispatchers must confer to decide the broader defining issues, such as which services need to be cancelled or re-routed, or reallocation of crew and rolling stock, etc. Based on these broader decisions, dispatchers then apply local-level approaches, described above for small scale disruptions, in the area under their supervision [2].

This research concentrates on the local-level remedies applied by local dispatchers for small-scale disruptions or as a second step after major disruptions. In this paper, the various remedies such as headway adjustment, or change in running order, are collectively referred to as ‘train-order rescheduling’.

2.2 Train-order rescheduling plan generation and related issues

Computer assisted train-order rescheduling generally involves setting a number of quantitative evaluation indices that will show the preferred proposed revised schedule, then combining various timetable changes and selecting the change that scores the highest preference index value. RTRI has so far been working on this basic approach to develop a train rescheduling algorithm. First, factors contributing to customer dissatisfaction were defined and quantified, such as train cancellation, reduction in services etc. Then, the meta-solution simulated annealing method [3], was used to identify the conditions that would minimize the overall score generated by these customer dissatisfaction criteria [4]. Even though this method can take into account delays and cancellations, it had several drawbacks: first, obtaining a result was time consuming, secondly there was the issue of objectivity in converting customer dissatisfaction into a numerical value.

To overcome these problems, another method based on mathematical optimization [5] was investigated [1, 6], which aimed to reduce the overall delay suffered by passengers in reaching their final destination, by changing only the order of trains. Mathematical optimization aims
to find the best solution (in this case, a train order rescheduling plan) within certain constraint conditions that are defined as a set of inequalities. The method then seeks to obtain the best evaluation index value from a set of evaluation indices that have been defined mathematically, which corresponds the best alternative train-order rescheduling plan. Research to apply this method to operation rescheduling is being carried out both in Japan and in other countries [7, 8].

Another method which uses constraint programming [9] has been used on Shinkansen train operation rescheduling [10].

This leading research means that computer assisted optimized train order rescheduling is now possible, in relation to a set of predefined evaluation indices. The output optimal solutions generated in this manner however, are only optimal for these defined (and restricted) conditions. In practice however, the conditions must cover all the constraints required in practical situations to get “absolutely optimal” proposals. To achieve this, the following two other problems must be solved.

The first problem relates to the limit on possible conditions taken into account. The data taken into account by the computer about the same object (a station or a train) will inevitably be different to all the factors considered by the dispatcher for decision making. For example, dispatchers may take into account time lags caused by a station’s facilities or communication means, the warning times required for particular track sections, or stations affected by disruptions on other lines etc. It is impossible to incorporate and keep updating all these factors in the system.

Another problem is related to quantification of conditions to consider. Dispatchers know instinctively, through experience, which conditions should be prioritized and to what degree. Interviewing dispatchers to shed light on this knowhow is not feasible, and so inserting such instinctive abilities in the form of quantitative constraints in an optimization method is not possible.

2.3 Approach adopted to tackle these problems

Based on the problems exposed above, this paper proposes a systematic approach train-order rescheduling, following the framework shown in Fig. 1.

First, a set of “rules” are created based on past dispatch decision data that reflect the instinctive knowhow underlying the dispatchers’ thought process. In the rules, trains and stations are generalized, that is, expressed as variables. The target area is delimited by specifying attributes and statuses. By matching the timetable status and the rules, alternative applicable rules and variables (trains, stations, etc.) are obtained and corresponding constraints are generated. Mathematical optimization therefore produces a rescheduling proposal that satisfies all the static, obvious constraints and the newly produced constraints mentioned previously. Details of algorithms described in section 4 are shown in Fig. 4.

This approach therefore generates the type of “absolutely optimal “train-order rescheduling proposal” mentioned earlier. The rules mentioned previously derived from dispatcher experience-based knowhow are therefore referred to as operational dispatching rules (ODRs).

Fig. 1 Framework for generating train-order rescheduling proposals

To realize this framework, the data expressing the generalized rules first has to be cleaned. Then, a mining algorithm to extract rules from the data on decisions made by dispatchers has to be created. These two steps are explained in the next section.

3. Operation dispatching rule

3.1 Operational dispatching rule requirements

In order to create ODRs, experience based intimate knowledge must be made explicit. To do so, ODRs must satisfy the following conditions so that they can be incorporated as constraints in mathematical optimization:

1. They must be made from static and dynamic data that can be referenced in the system.
2. They must be composed of conditions that the train diagram must satisfy after rescheduling (X), and conditions (Y) that satisfy the rescheduled timetable.

The ODR would therefore satisfy the structure, “If the current timetable and traffic forecasts satisfy condition X, then the timetable must be modified to satisfy condition Y”, which using the terminology of association rule mining, described later, would be expressed as X is the antecedent and Y is the consequent.

Each part is structured like the matrix in Table1, reduced from event diagrams, which are explained in the next section.

3.2 Event diagram

Events that are the basis for the rules in the timetable are represented by different forms in the event diagram, e.g. departure and arrival events are represented by nodes, the sequential relationship between two events is represented by an arc linked from the preceding event to the subsequent one. An arc representing the scheduling of arrivals and departures influenced by signaling and route control schedules is referred to as the control arc. An arc representing the scheduling of different events for a same train is referred to as the train arc.

Rescheduling the order of train departures corresponds
to altering the event diagram: a change in departure platform will lead to a change in a node which is the platform attribute while modifying the train order will add or delete an arc.

Figure 2 illustrates train-order rescheduling, where the dispatcher has changed the order of arrival of outbound train No. 1 from and inbound train No. 4 to station B to avoid the foreseeable extra delay that train No. 1 would cause for train No. 4. The characters on the node in Fig. 2 represent events, where “1.A.a” means that train No. 1 arrives at Station A at the station only after the arrival of the inbound train.

The dashed gray lines represent the original or rescheduled timetable, while solid green lines overwritten on the original timetable represent the predicted state of operations. The transition in the predicted situations shown in Fig. 2 can be described as the event diagram shown in Fig. 3. Each solid arc represents the relationship in the order of events for the same train, while each dashed arc represents the controlling order of the different trains. The colored nodes signify events predicted to cause a delay because of the propagation of a previous delay. Comparing the transition in the network topography between two diagrams, reveals that only the direction of one dashed arc is different: the arc between node 1.B.a and node 4.B.a, which corresponds to the two events, train No. 1’s arrival to station B and train No. 4’s arrival to station B. It is also possible to see that train No. 4’s delay would be resolved with this change in train order change because the delay to train No. 1 would not influence train No. 4.

3.3 Data structure of ODR based on event diagram

The components of ODR based on the event diagram are as follows:
(1) A set of node attributes required as antecedent in the event diagram before rescheduling.
(2) A set of arcs required as antecedent in the event diagram before rescheduling.
(3) A set of modified node attributes following rescheduling.
(4) A set of modified arcs following rescheduling. Every modification to an event diagram corresponding to a single operation change in the timetable, can be described by the above structure.

The rescheduling in Fig. 2 is assumed to have been achieved following the ODR as follows:
Antecedent: If the delay in the arrival event of an outbound train at station B results in more than a 5 minute delay of the arrival of an inbound train,
Consequent: then the order between such two events shall be changed, that is, the outbound train is made to arrive at the station only after the arrival of the inbound train.

The components of this ODR are shown in Table 1.

| Table 1 An example ODR components |
|-----------------------------------|
| (1) Node attributes of antecedent  |
| Node | Line | Station | Track | Delay |
| x    | Outbound | B      | Tr. 2 |      |
| y    | Inbound  | B      | Tr. 3 | 5 min.|
| (2) Arc component of antecedent   |
| Head | Tail | Type | Delay propagation |
| x    | y    | Different | Yes |
| (3) Node attributes modified as consequent  |
| Node | Line | Station | Track |
| (4) Arcs modified as consequent |
| Head | Tail | Type | Modification type |
| x    | y    | Different | Deletion |
| y    | x    | Different | Addition |

4. Mining of ODR rules

4.1 Association rule

Depending on the dispatcher, the decision taken for a same situation may be different because of their differing experience and knowhow. Therefore, it was considered that ODRs should be extracted as association rules [11] obtained by statistically analyzing previous records of train operation rescheduling. A general association rule is of the form $X \Rightarrow Y$. $X$, $Y$ are sets of items. In other words, the elements of $X$ and $Y$ are called “items” in a general association rule which is the approach adopted in this paper. $X$ is the
antecedent. \( Y \) is the consequent. A rule is defined as \( X \rightarrow Y \), on a set of transactions \( D = \{ s_1, s_2, \cdots, s_J \} \) stored in what is called the database, where element \( s_i \) is also a set of items. The form means that if a transaction \( s_i \) in \( D \) contains all items of \( X \), \( s_i \) also contains all items of \( Y \).

There are two criteria called ‘support’ and ‘confidence’ used to select reasonable association rules. The support for a rule \( X \rightarrow Y \), with respect to the database \( D \), is the proportion of transactions in \( D \) which contain \( X \) and \( Y \). The confidence of a rule \( X \rightarrow Y \), with respect to the database \( D \), is the proportion of the transactions that contain \( X \) which also contain \( Y \).

The definition of the association rule above needs to be slightly refined in order to express the time lag between the items in the ODRs. In the refined definition, each element from database is a sequence \( s_i = (t_{i1}, t_{i2}, \cdots, t_{im}) \), where \( t_{ij} \) is a set of items. Hereinafter, ‘transaction’ is used to describe an element \( t_{ij} \) of sequence \( s_i \). The antecedent \( X \) of each rule is composed of a set of items. And the consequent \( Y \) of each rule is a sequence of transactions. Mining rules according to this definition is called sequential rule mining [12].

4.2 Collection of ODR elements from the database

For each type of local rescheduling remedy, such as change in train order, and subsequent change in platform, etc., data concerning the conditions accompanying each rescheduling operation and how the timetable was modified, need to be extracted from the database to form the basis of the rules. Consequently, data series that fulfilled the following condition: \( s = (t_1, t_2, \cdots, t_m) \), were defined as the elements in the target data base, and the relevant data was collected from past operations records.

1. A pre-defined maximum limit is set for the number (\( N \)) of events in a rule, and all events in a rule \( s \) make up some connected part of the whole event diagram representing the whole train timetable.
2. Each transaction \( t_j \) made up a set of items, corresponds to an event diagram at a moment for \( N \) events, and contains all the attributes of the nodes and the arcs. However, it include at least one unrealized event (forecast event) at the moment.
3. The order of the transactions represents the order of changes to the timetable starting from the event diagram \( t_1 \) which is delayed.

4.3 Steps for mining ODR rules

1. Classification of database elements based on network topography

It is possible to classify dispatching rules according to the network topography of the antecedent event diagram. As such, starting with the network topography \( p \) resulting from the number of nodes \( N \), the topographies of the event diagram \( t_1 \) resulting from early database elements that correspond with \( p \) are extracted, to make a database \( D_r \). This makes it possible to reduce the volume of target data and cut the processing time.

2. Frequent sequential pattern mining

Before extracting a rule \( r_1 \rightarrow (r_2, r_3, \cdots, r_i) \), where \( r_i \) is a set of items, the frequent sequential pattern \( R = (r_1, r_2, \cdots, r_i) \) is extracted from the database by using an Prefix Span [13] algorithm adapted for pattern mining. The Prefix Span is a general extraction algorithm, and can extract a pattern \( R \) where \( \text{Supp}(R) \) is greater than or equal its minimum threshold \( \text{Supp}_{\text{min}} \). Here, \( \text{Supp}(R) \) is defined as the number of elements \( s = (t_1, t_2, \cdots, t_m) \) in the database which satisfy the two conditions \( R \subset s \) and \( t_1 \subset t_l \), so that antecedent \( r_1 \) must be selected from the first event diagram \( t_1 \).

3. Determination of association rule validity

An extracted pattern \( R \) which has a length greater than or equal to 2, can be an association rule if and only if its confidence \( \text{Conf} (r_1 \rightarrow (r_2, \cdots, r_i)) = (\text{Supp}(R))/\text{Supp}(r_i) \) is greater than or equal to the minimum threshold \( \text{Conf}_{\text{min}} \). ODRs are extracted on the basis of this condition, to determine validity.

5. Proposal for train-order rescheduling method reflecting extracted ODR rules

The method developed to generate revised train-order scheduling plans reflects the operational dispatching rules that were extracted by the above-mentioned extraction method. The method was built using a combination of data-mined dispatching rules and the existing rescheduling proposal generating method based on mathematical optimization. Figure 4 illustrates the steps in the method’s process.

First, an initial proposal is made on the basis of the existing unmodified timetable. Then, looking at the event diagram generated from the first proposal, if there are any points that match the antecedent of a rule, the corresponding consequent is adapted and a constraint condition is added again. Under the new constraint condition, the next step seeks to find an absolutely optimized solution that...
surpasses the optimized solution that can be found using the existing mathematical optimization method. The process goes back again to step one to find out if there are any matches with existing rules. In order to find the absolutely optimized solution, the process is repeated until there are no more matches with the antecedent.

When the ODR matches an antecedent, the constraints that match the corresponding consequent are added to the event diagram only for the earliest point in the antecedent. The rescheduling plan is updated using conventional mathematical optimization each time after this step is completed. This approach reduces the number of times the process has to be repeated by gradually working through all the matching spots chronologically.

6. Confirmation tests

6.1 Test ODR extractions

Tests were carried out to extract ODRs using three sets of train-order rescheduling records for a section of single track. All the rescheduling plans contained 90 modifications. The minimum support was set to 3 times (i.e., 3 modifications in all) and minimum confidence to 1.0. When the number of events N as set to 3, 31 rules were extracted after about 1.5 minutes of calculation. When N was set to 4 however, 170 rules were extracted after about 54 minutes. A normal personal computer with 64 bit, 7 core CPU running Windows10 was used to carry out the calculations. In the case where N=3, the content of the 31 extracted rules was analyzed. It was found that some identical rules presented a different event diagram topography, so there were in fact 15 essentially different rules. The reason for the duplication was found to be that unrelated events had been included in the rule’s event diagram. This repetition can be avoided by eliminating such events when generating rules. The significantly higher number of rules extracted when N=4 is considered to have been the result of the same duplication problem.

The following two rules were selected from among the extracted rules for the next test.

(Rule 1)
Antecedent: If on a single track section between stations P and Q, the order of trains is first outbound train ‘a’ followed by inbound train ‘b’ and the delay between the arrival and departure of train ‘b’ at station Q is greater than 5 minutes and less than 15 minutes,

Consequent: then (1) the train order shall be changed, and (2) arrival track of train ‘a’ at station P shall be changed to platform 2.

(Rule 2)
Antecedent: If the train order on a single track section between stations H and I is first outbound train ‘a’, followed by inbound train ‘b’, and inbound train ‘c’, and the delays between the arrival and departure of both trains ‘b’ and ‘c’ at station I are greater than 5 minutes and less than 15 minutes,

Consequent: then the train order shall be changed to train ‘b’, train ‘c’, and train ‘a’.

Fig. 5 Overview of Rule 1

Rule 1 illustrates the case where a station’s outbound platform must be changed to adapt to the modified train order in order to avoid conflicting need to use the same platform as shown in Fig. 5. Rule 2 illustrates a special case where the order of 3 trains is changed.

6.2 The test to generate train-order rescheduling proposals

To check the cases that would suit Rule 1, two hypothetical delay scenarios were constructed, and the conventional method and new algorithm containing Rule 1 were applied. The resulting train order rescheduling proposals with both methods prioritized the inbound train. Under the conventional method, the platform for the inbound train was changed in a number of case, to take into account the modified train order, while with the new algorithm, it was the platform for the outbound train that was changed to take into account the modified train order. In practice, however, the proposal from the conventional method may not be suitable, because it was inconsistent with Rule 1.

In tests to apply Rule 2, one of the three cases taken from the data records to extract the rule was used, and both the conventional method and the new algorithm were applied. The results of the tests using both methods and the rescheduling proposal made by an actual dispatcher are shown in Fig. 6. The orange lines show the actual operation rescheduling proposal made by the dispatcher, while the two green lines correspond to the inbound trains.

Fig. 6 Comparison of proposals
and the two blue lines correspond to outbound trains from the proposals generated by each method, superimposed over the orange lines. The yellow lines in each proposal represent the parts matching the antecedent of Rule 2. The two dashed red reference lines reveal that the recovery of the delay of the inbound train was faster with the actual dispatcher’s proposed rescheduling plan than the plan generated with the conventional method, inversely the outbound train delay recovery was relatively slower with the dispatcher’s plan than with the plan produced by the conventional method. The same tendency was found in the plan generated using the new method. In terms of total delay time, however, which was the optimization criterion in both the conventional and the developed methods, the plan generated using the conventional method was best. This indicates that proposals prioritizing the delay recovery of specific trains rather than overall timetable recovery can be generated by implementation of ODRs.

7. Conclusions

A method was developed to reflect specialist dispatcher knowledge and know-how which based on computer generated operation rescheduling plan proposals. This paper explains the method for extracting ODRs, the train-order rescheduling proposal generation process, and presents the results of functional confirmation tests.

Work will continue to resolve new problems that were identified during ODR extraction tests, such as duplication of rules. In addition, the confidence of extracted rules could not be confirmed because of the insufficient volume of train-order rescheduling records used for the tests. Plans for future work include reviewing the confidence of ODRs and performance of the method by using big data.

The developed method does take into account local conditions specific to line sections or stations. Additional work will also be carried out on the interface, to allow dynamic generation of constraint expressions from ODRs which is required for practical use.

References

[1] Sato, K. and Hirai, C., “Timetable Rescheduling Formulation and Algorithm Minimizing The Total Increase of Inconvenience to Passengers,” RTRI Report, Vol. 30, No. 8, pp. 11–16, 2016 (in Japanese).

[2] The Survey Special Committee of the Inst. of Electrical Engineers of Japan on Enhancement of Tasks of Operation Scheduling and Operation Management on the Railroad (Eds.), Tetsudo daiya kaifuku no gijutsu (Recovery Technique on Railroad Timetable), Ohmsha Ltd., Japan, pp. 16–17, 2010 (in Japanese).

[3] Yanagura, M. and Ibaraki, T., Kumiaiwasu saitekika : meta senryaku o chushin to shite [Combinational Optimization : Focusing on Meta Strategy], Asakura Publishing Co., Ltd., Japan, 2001 (in Japanese).

[4] Tomii, N., Tashiro, Y., Tanabe, N., Hirai, C. and Muraki, K., “Train Rescheduling Algorithm which Minimizes Passengers’ Dissatisfaction,” IPSJ Transactions on Mathematical Modeling and its Applications, Vol. 46, No. SIG-2, pp. 26–38, 2005 (in Japanese).

[5] Fujisawa, K. and Umetani, T., (Kuno, K., Tamura, A. and Matsui, T. Eds.), Ouyou ni yakudatsu 50 no saitekika mondai [Fifty Optimization Problems Useful for Application], Asakura Publishing Co., Ltd., Japan, 2009 (in Japanese).

[6] Sato, K., Tamura, K. and Tomii, N., “A MIP-based timetable rescheduling formulation and algorithm minimizing further inconvenience to passengers,” Journal of Rail Transport Planning & Management, Vol.3, pp. 38–53, 2013.

[7] Imada, K. and Tomii, N., “Rescheduling Algorithm Based on MILP Formulation Considering Partial Cancellation and Turning Back,” The transactions of the Institute of Electrical Engineers of Japan. D, Vol. 137, No. 6, pp. 484–491, 2017 (in Japanese).

[8] Dollevoet, T., Huisman, D., Schmidt, M. and Schobel, A., Delay management with rerouting of passengers, Transportation Science, Vol.46, No.1, pp.74-89, 2012.

[9] Mizoguchi, F., Furukawa, K., Lassez, J.-L. (Eds.), Seiyaku ronri programming [Constraint Logic Programming] (Supervising Fuchi, K.), Kyoritsu Shuppan Co., Ltd., Japan, 1989 (in Japanese).

[10] Shimizu, H. and Nozue, N., “Atarashii keikaku gijutsu to tetsudou unkou kanri : Seiyaku programming o mochita Shinkansenen unten seiri system [New Scheduling and Railroad Operation Management : Shinkansen Operation Management System by Using Constraint programming],” IIE-Japan Industry Applications Society Conference, No. S7-4, pp. 831–836, 2002 (in Japanese).

[11] Kitsuregawa, M., “Mining Algorithms for Association Rules,” Journal of Japanese Society for Artificial Intelligence, Vol. 12, No. 4, pp. 513–520, 1997 (in Japanese).

[12] Kour, A., “Sequential Rule Mining, Methods and Techniques: A Review,” International Journal of Computational Intelligence Research, Vol.13, No.7, pp.1709-1715, 2017.

[13] Sharma, P. and Balakrishna, G., “PrefixSpan: Mining sequential patterns by prefix-projected pattern,” International Journal of Computer Science & Engineering Survey, Vol.2, No.4, pp.111-122, 2011.

Authors

Takashi SAKAGUCHI
Senior Chief Researcher, Head of Transport Operation Systems Laboratory, Signalling and Transport Information Technology Division
Research Areas: Railroad Operation Scheduling, Combinational Optimization, Operation Management technologies

Keisuke Sato, Ph. Dr.
Assistant Senior Researcher, Transport Operation Systems Laboratory, Signalling and Transport Information Technology Division (Currently Kanagawa University)
Research Areas: Railroad Operation Scheduling, Mathematical Optimization