Reliability and Delay Distributions of Train Connections*

Mohammad H. Keyhani, Mathias Schnee, Karsten Weihe, and Hans-Peter Zorn

Darmstadt University of Technology, Computer Science,
Hochschulstraße 10, 64289 Darmstadt, Germany
{keyhani,schnee,weihe,zorn}@algo.informatik.tu-darmstadt.de

Abstract

Finding reliable train connections is a considerable issue in timetable information since train delays perturb the timetable daily. We present an effective probabilistic approach for estimating the reliability of connections in a large train network. Experiments on real customer queries and real timetables for all trains in Germany show that our approach can be implemented to deliver good results at the expense of only little processing time. Based on probability distributions for train events in connections, we estimate the reliability of connections. We have analyzed our computed *reliability ratings* by validating our predictions against real delay data from German Railways. This study shows that we are able to predict the feasibility of connections very well. In essence, our predictions are slightly optimistic for connections with a high rating and pretty accurate for connections with a medium rating. Only for the rare cases of a very low rating, we are too pessimistic.

Our probabilistic approach already delivers good results, still has improvement potential, and offers a new perspective in the search for more reliable connections in order to bring passengers safely to their destinations even in case of delays.

1998 ACM Subject Classification G.2.2 Graph Theory (Graph algorithms; Network problems)

Keywords and phrases Stochastic Delay Propagation, Timetable Information, Connection Reliability

Digital Object Identifier 10.4230/OASIcs.ATMOS.2012.35

1 Introduction and Motivation

Timetable information systems have the ability to find attractive train connections according to criteria such as travel time, number of transfers, price, etc. The reliability of the connections plays a crucial role since the timetable continually gets perturbed because of delays of trains. Connections, which were found according to the timetable, may get infeasible if a scheduled transfer is no longer possible due to arriving too late for the transfer.

State-of-the-art commercial systems predict the arrival and departure times of trains by computing scalar delay values given in minutes. Consequently, the reliability of connections can be rated based on only one possible delay value for each departure and arrival event in the connection. The drawback of this approach is that the predicted delays often deviate from the actual delays. An obviously better approach, which we present in this paper, is to consider all possible delay values weighted by the probability of occurrence. For each departure and arrival event of the trains in a connection, we calculate for each possible

* This work was partially supported by German Railways Deutsche Bahn AG (RIS).

12th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS'12).
Editors: Daniel Delling, Leo Liberti; pp. 35–46

OpenAccess Series in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

delay value the probability that the train has this delay. These probability distributions are calculated based on timetable data, latest available delay data, and waiting time policies for transfers between trains.

Our Contribution. We introduce the reliability rating *rel*, which scores the reliability of a connection in percentage terms. Our probabilistic approach allows us to calculate delay distributions for connections and to reasonably estimate their *rel*-rating in order to advise passengers against choosing connections tending to break and guide them towards more robust ones. To our knowledge, we are the first to extend distributions for departure and arrival events of trains to explicitly model the reliability of transfers and connections. In this paper, we will not only present the mathematical formula to calculate these distributions but also a computational study which demonstrates promising run-time behavior and good quality of the results of a prototype implementation. In the outlook in Section 5.2, we will also mention how we plan to use these distributions to improve the search for reliable connections in our existing multi-criteria timetable information system MOTIS [5].

Related work. Delay propagation and prediction has been studied by means of deterministic and stochastic models as well as simulations, especially in the field of decision support for network dispatchers and timetabling. Experiments with a deterministic model by Müller-Hannemann and Schnee showed that timetables can be updated with a large amount of delay and forecast data in real time to allow for up-to-date timetable information. They continuously adjusted their graph representing the schedule according to the real-time data to always represent the current situation. In their multi-server architecture each timetable information server only spends 0.1% of the day with updating and maintenance [6]. Simulations are the basis of the predictions by Lu et al. [3] for various network topologies (single and multi-track) and Murali et al. in [7]. The latter estimated delays for freight trains only.

Meester and Muns used so-called *phase-type distributions* in their model for stochastic delay propagation in railway networks in [4]. Carey and Kwieciński also use approximations in their model [2]. However, in those papers waiting policies are not respected. A nice overview of models can be found in Yuan's PhD thesis [8]. The stochastic model which comes really close to our approach is due to Berger et al. [1]. They basically have the same model for train distributions and also respect waiting policies. However, they concentrate on trains, not on entire connections, and do not investigate reliability. We will enhance their formulas to calculate probability distributions for connections consisting of several trains and transfers between them.

Overview. This paper is organized as follows. In the next section, we will briefly introduce the timetable data and operational concepts. In Section 3, we will describe our probability distributions and how we calculate them in detail. Our experiments and computational results will be reported on in Section 4. Finally, we conclude and present an outlook on our future work.

2 Train Operation

The timetable. For our work we use real-world timetables without simplifying assumptions. The timetable is the current timetable of German Railways Deutsche Bahn AG. Besides the scheduled times for arrival and departure events we also respect the transfer times required to change trains (dependent on the size of the station and the platforms the trains stop at). There is also the possibility to walk a short distance from one station to another, e.g. from a

main station to its smaller local train station, called a *footpath*.

Waiting policies. In daily operations, a set of policies, the *waiting time rules*, describes the maximum amount of time a train will wait to allow passengers a transfer from a delayed feeder. Each train may have a number of feeders with different applicable waiting times. The connecting train will leave delayed if one of its feeders is delayed and the arrival time plus the required transfer time from the feeder is not later than the scheduled departure time plus waiting time.

Real-time data. We constantly receive current delay data for German trains in a live-feed from Deutsche Bahn AG. This delay data is integrated into our representation of the timetable and used to update our probability distributions. These messages state that a train has arrived or departed at a certain point in time (either on time or delayed), and are denoted as *is-messages*.

3 Probability Distributions

3.1 Our Model

A timetable $TT := (TR, S, EC)$ consists of a set of trains TR, a set of stations S, and a set of elementary connections EC. Each $ec \in EC$ is defined by its events dep_{s_1} and arr_{s_2} corresponding to the departure event at station $s_1 \in S$ and the arrival event at station $s_2 \in S$. Each train $tr \in TR$ consists of a set of successive elementary connections ec_i, where the arrival event of ec_i and the departure event of ec_{i+1} are at the same station. Let DEP be the set of all departure events, ARR the set of all arrival events, and $EVENTS := DEP \cup ARR$. For each event $event \in EVENTS$, $sched(event) : EVENTS \mapsto \mathbb{N}$ is the scheduled time-stamp of the event given in minutes. A delay $d \in \mathbb{Z}$ is the difference between the scheduled time-stamp and the actual time the event occurs. According to a policy in German Railways operation no train is allowed to depart before its scheduled departure time. Therefore, departure delays are non-negative.

The minimal standing time $stand(tr, s)$ defines how long train tr has to wait at station s after its arrival and before its departure. The necessary transfer time from a train $tr_1 \in TR$ into another train $tr_2 \in TR$ is denoted by $transfer(tr_1, tr_2)$. According to the waiting time rules, the maximal waiting time of tr_2 for tr_1 at station $s \in S$ is defined by $wait(tr_2, tr_1)$. At a given station s, a train $f \in TR$ is a potential feeder for another train $tr \in TR$ if a transfer from f into tr is possible, tr would wait for f for at least 1 minute according to the waiting time rules, and the difference between the scheduled departure time of tr and the scheduled arrival time of f is not greater than a given parameter γ. Currently, we use $\gamma = 30$, since the transfer times are at most 20 and the waiting times at most 10 minutes. For each departure event $dep_{tr,s}$ of train tr at station s there exists a set of feeder trains $FD(tr, s) \subset TR$. The maximal waiting time of tr_2 at station s for any feeder is defined by $wait_{max}(tr, s) := \max_{f \in FD(tr,s)}\{wait(tr, f)\}$.

Let (Ω, A, P) be a discrete probability space with sample space Ω, σ-algebra A, and probability measure P. We use discrete random variables $X : \Omega \mapsto \mathbb{N}$ for mapping train events to time-stamps. We define the discrete random variable $X_{event} : \Omega \mapsto \{sched(event), sched(event) + 1, \ldots\}$ which is the actual time of $event \in EVENTS$ given in minutes.

We assume that the distributions of the arrival times of all feeder trains of a given train are stochastically independent. This assumption does not hold for all feeder trains, especially if two feeders have a common feeder or are disturbed by a common reason (e.g. a problem at

a certain track). The derived delay distributions may be biased as conjectured by Meester and Muns [4]. This fact warrants further investigation.

Input distributions. For each elementary connection ec of train tr from dep_{tr,s_1} to arr_{tr,s_2}, there is a set $X_{travel} = \{X^d_{travel} \mid d \in \mathbb{N}\}$ of probability distributions for the travel time. X^d_{travel} is the conditional distribution of the travel time of ec given a departure delay $d \in \mathbb{N}_0$ in minutes. They represent the potential of making up for the current delay and the possibility of further delays on ec. We generate these travel time distributions depending on the scheduled travel time of the elementary connections.

3.2 Distributions for Connections

For each train event, we have already defined the scheduled time *sched(event)*. In fact, the actual time of an event could be shifted according to delays. We intend to predict the delay of an event by analyzing the time interval in which the event could take place. For each minute in this interval, we determine the probability that the train event actually occurs at this point in time. Hereby, a probability distribution arises and can be used as a prediction for the event time. In this section, we explain in detail how probability distributions of train connections are calculated.

Definition of connection. A connection c defines a feasible path $s_1, s_2 \dots s_n$ between a start station s_1 and a target station s_n by a set of successive elementary connections $EC_c = \{ec_1, ec_2, \dots\} \subset EC$. Two successive elementary connections ec_i and ec_{i+1} either belong to the same train tr, or to two different trains tr_1 and tr_2. In the second case, there is a feasible transfer between tr_1 and tr_2 at the corresponding station s. The difference between the departure time of train tr_2 and the arrival time of train tr_1 at station s is greater than or equal to the required transfer time between these two trains, which is denoted by $transfer(tr_1, tr_2)$. There also exists a special case of transfers, where after leaving train tr_1 at station s_1 a footpath is used in order to walk to another station s_2 and to enter the departing train tr_2. In this case, the required walking time is used as the required transfer time between the two trains. A connection is denoted as *direct connection* if all elementary connections EC_c belong to the same train.

Definition of probability distribution. Let $t_{start}, t_{end} \in \mathbb{N}$ be two timestamps defining the bounds of a time interval. The probability distribution of a departure event is determined by calculating the probabilities $P\big(X_{dep} = t\big)$ for all $t \in [sched(dep_{tr_2,s}), t_{end}]$. The right bound t_{end} of the interval is chosen so that there is no time $t > t_{end}$ with $P\big(X_{dep} = t\big) > 0$. The probability distribution of an arrival event is determined by calculating the probabilities $P\big(X_{arr} = a\big)$ for all $a \in [t_{start}, t_{end}]$, whereby the bounds are chosen so that there is no time $a \notin [t_{start}, t_{end}]$ with $P\big(X_{arr} = a\big) > 0$. We denote a distribution of an event by $pd(event) \subset \mathbb{R}^w$ where $w = (t_{end} - t_{start}) + 1$, and define it as a tuple of probabilities according to all minutes in the corresponding time interval. The distribution of a connection $pd(c)$ is equal to the distribution of its last arrival event.

3.2.1 Calculation of Distributions

Considering $EC_c = \{ec_1, ec_2, \dots\}$, starting with the first departure of the connection, we calculate the distribution of each event until we reach the last arrival. We have to distinguish between departures with and without transfer at the station. In this section, we explain in detail how the probability distribution for a departure event after a transfer is calculated. Then, we will mention how this approach is modified for the other cases. Figure 1 illustrates a

departure of train tr_2 after a transfer from train tr_1, with $FD(tr_2, s) = \{f_1, f_2\}$. Theoretically, tr_1 could also be a feeder of tr_2. In that case, it is treated separately in some of the formulas and not together with the other feeders. Recall that a train has to wait at a station for a minimal standing time after its arrival to allow boarding and leaving the train. Therefore, the distribution of the departure event $dep_{tr,s}$ depends on the preceding arrival $arr_{tr,s}$, on the set of its feeders $FD(tr, s)$, and in case there is a transfer into tr_2, also on the arriving train tr_1. The feasibility of the transfer only depends on whether or not the transfer time $transfer(tr_1, tr_2)$ from tr_1 to tr_2 is satisfied. Note that the feeders can only introduce additional delays.

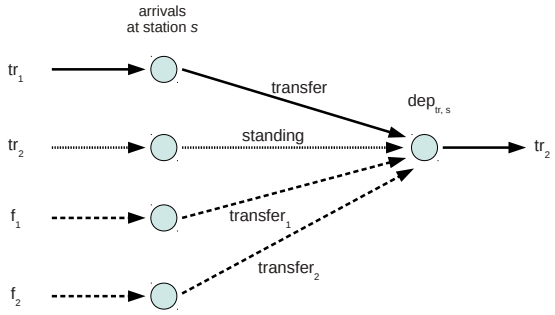

Figure 1 Departure of train tr_2 after a transfer from train tr_1.

Considering the departure event $dep_{tr_2,s}$ after a transfer from train tr_1, now, we are able to calculate the probability distribution of this event. The departure takes place in the interval $[sched(dep_{tr,s}), t_{end}]$. We distinguish between these cases:

1. Train tr_2 departs at its scheduled time $sched(dep_{tr,s})$.
2. Train tr_2 departs at time $t \in [sched(dep_{tr,s}) + 1, sched(dep_{tr,s}) + wait_{max}(tr, s)]$. In this time interval the train may have to wait for its feeders.
3. Train tr_2 departs at $t \in [sched(dep_{tr,s}) + wait_{max}(tr, s) + 1, t_{end}]$. In this time interval the train does not have to wait for any feeder.

In all three cases, a feasible transfer from tr_1 to tr_2 has to be ensured. In the following, we present the formulas to calculate the probabilities for the minutes of each subinterval.

3.2.1.1 Departing at the scheduled time

A departure at time $t = sched(dep_{tr_2,s})$ is possible if

- tr_2 arrives at time $t_2 \leq t - stand(tr_2, s)$,
- tr_1 arrives at time $t_1 \leq t - transfer(tr_1, tr_2)$,
- and tr_2 does not have to wait for any other feeder.

We use this formula to calculate the probability:

$$P\big(X_{dep} = t\big) = P\big(X_{arr_{tr_2,s}} \leq t - stand(tr_2, s)\big) \cdot P\big(X_{arr_{tr_1,s}} \leq t - transfer(tr_1, tr_2)\big) \cdot P_{noWaitingForFeeders}\big(tr_2, s, t\big)$$

The term $P_{noWaitingForFeeders}\big(tr_2, s, t\big)$ corresponds to the probability that the train tr_2 does not have to wait for any other feeder. The formula is omitted due to space restrictions.

3.2.1.2 Departing within the waiting interval

Train tr_2 departs delayed at time $t \in [\mathit{sched}(\mathit{dep}_{tr_2,s}) + 1, \mathit{sched}(\mathit{dep}_{tr,s}) + \mathit{wait}_{max}(tr, s)]$ in one of the following cases:

1. The delayed departure at time t is because of a delay of $\mathit{arr}_{tr_2,s}$. This happens if
   - tr_2 has a delay and arrives exactly at time $t_2 = t - \mathit{stand}(tr_2, s)$,
   - tr_1 arrives at time $t_1 \leq t - \mathit{transfer}(tr_1, tr_2)$,
   - and tr_2 does not have to wait longer for any other feeder.
2. The delayed departure at time t is only because of waiting time rules. This happens if
   - tr_2 arrives at time $t_2 < t - \mathit{stand}(tr_2, s)$,
   - tr_2 has to wait for tr_1 or for at least one of the other feeders. This probability is denoted by $P_{\mathit{waiting}}(tr_2, s, t)$ (formula omitted due to space restrictions).

We use this formula to calculate the probability:

$$\begin{aligned} P\big(X_{dep} = t\big) = P\big(X_{arr_{tr_2,s}} = t - \mathit{stand}(tr_2, s)\big) \cdot P\big(X_{arr_{tr_1,s}} \leq t - \mathit{transfer}(tr_1, tr_2)\big) \\ \cdot P_{\mathit{noWaitingForFeeders}}(tr_2, s, t) \\ + P\big(X_{arr_{tr_2,s}} < t - \mathit{stand}(tr_2, s)\big) \cdot P_{\mathit{waiting}}(tr_2, s, t) \end{aligned}$$

3.2.1.3 Departing after the waiting interval

Train tr_2 departs at time $t \in [\mathit{sched}(\mathit{dep}_{tr,s}) + \mathit{wait}_{max}(tr, s) + 1, t_{end}]$ if

- tr_2 is delayed so that it does not have to wait longer for any feeder,
- and tr_1 arrives at time $t_1 \leq t - \mathit{transfer}(tr_1, tr_2)$.

To calculate this probability, we simplify the previous formula as follows:

$$P\big(X_{dep} = t\big) = P\big(X_{arr_{tr_2,s}} = t - \mathit{stand}(tr_2, s)\big) \cdot P\big(X_{arr_{tr_1,s}} \leq t - \mathit{transfer}(tr_1, tr_2)\big)$$

By applying the above formulas, we are able to calculate the distribution for a departure after a transfer. Distributions for normal departure events without transfers can be obtained by modifying these formulas. Since there is no train tr_1 anymore, we only have to consider the train itself and its feeders. When a departure is the first departure event of a train, the arrival time of the train at the station is ignored.

3.2.1.4 Arriving at a given time

The probability distribution of the arrival time depends on the distribution of X_{dep} and the corresponding X^d_{travel} distributions. We obtain the probability $P\big(X_{arr} = a\big)$ analogous to the Bayes' theorem:

$$P\big(X_{arr} = a\big) = \sum_{d=0}^{a} P\big(X^d_{travel} = a - d\big) \cdot P\big(X_{dep} = d\big).$$

3.2.1.5 Probability of connection break

To calculate the distribution of our connection, we only consider the cases in which all transfers in the connection are feasible. These probabilities sum up to 1 if there are no transfers in the connection or if the transfers are feasible in all possible scenarios. After each transfer, this sum may decrease if a connection break is possible. For each distribution pd, we define the probability that the connection is not feasible: $P_{broken}(pd) = 1 - \sum_{t \in [t_{start}, t_{end}]} P\big(X_{dep} = t\big)$.

3.2.1.6 Treatment of is-messages

When distributions for events in the past are calculated, it may happen that we already have received a real-time is-message for an event so that the actual time is already known. In this case a one-point distribution can be used: $pd(event) = \{0 \dots p \dots 0\}$, where the probability p equals $1 - P_{broken}(pd)$ and corresponds to the known actual time of the event.

3.2.2 Reliability-Rating of a Connection

The sum of the calculated probabilities of the last arrival event, excluding $P_{broken}(pd)$, equals the probability that the connection is feasible. It can be used to rate the reliability of the connection and is defined as $rel(c) = 1 - P_{broken}(pd)$.

3.3 Distributions for Trains

We have already mentioned that, to calculate the distribution of an event, the distributions of all preceding events have to be known. For a departure event we need the arrival distributions of all involved feeders and if there is a transfer at the station also the arrival distribution of the arriving train we want to change from. All other required distributions will be calculated according to our approach introduced above. We calculate the probability distributions of the train events with the same formulas which we use for the events of connections, whereby there are no transfers over the course of trains. Our approach to calculate probability distributions for train events is similar to the approach presented in [1]. Since it would be very inefficient to calculate the distributions of all involved trains for every connection, we calculate for all train events in the timetable an initial probability distribution at the beginning of the day. Whenever an is-message for an event is received, its distribution is replaced by a one-point distribution by setting the probability of the actual event time to 1. Then the distributions of all of its succeeding events are recalculated. Recursively, for each of these events the distributions of all their succeeding events have to be recalculated. To restrict the number of affected nodes, if the distribution of an event changes only negligibly we do not recompute the distributions of its succeeding events. In order to keep the timetable up to date, is-messages are introduced into the graph every minute on each day.

4 Computational Study

4.1 Setup

Our computations were carried out on different desktop PCs with Pentium i5-2400 quad-core CPUs and 16 GB of RAM. We prepared time-expanded graphs for a number of two-day periods[1] as used for our multi-criteria timetable-information system MOTIS when taking delays into account [6]. A feeder edge is introduced if a train f is a potential feeder for another train t, the difference between arrival of f and departure of t is at most γ minutes, and a waiting time rule applies between f and t at the station. Currently, we use $\gamma = 30$ minutes. The graphs have between 1.7M event nodes, 0.9M train edges, and 80k feeder edges (smallest graph for Saturday and Sunday), 1.9M - 2.0M event nodes, about 1.0M train edges, and 94-100k feeder edges (Sunday and Monday *SuMo*, respectively Friday and Saturday *FrSa*) and 2.2M event nodes, 1.1M train edges, and 111k feeder edges (weekday only graphs).

[1] A two-day period is needed to cover long running trains and overnight connections

Table 1 Run-times and numbers of processed messages for updating train distributions with real-time information from is-messages.

Day	Messages		Run-time		
	(total)	(per min)	(total)	(per min)	peak
Monday	454,325	315	154.88s	108.0ms	630ms
Tuesday	436,379	303	150.25s	104.6ms	540ms
Wednesday	399,073	277	140.15s	97.5ms	560ms
Thursday	436,142	302	161.68s	112.5ms	790ms
Friday	432,574	300	157.49s	109.6ms	650ms
Saturday	431,531	299	145.05s	101.0ms	620ms
Sunday	405,161	281	140.79s	98.0ms	610ms

4.2 Computational Results

Initial distributions. For three weeks in June 2012, we repeatedly calculated the initial distributions and averaged over three runs per day. The average time required on weekdays only is 74.7s, for SuMo and FrSa graphs 65.0s resp. 67.7s and for weekend graphs 57.8s. Note that in daily operations these computations can be executed beforehand and read from a file at start-up.

Real-time update for train distributions. The run-times and number of processed is-messages for updating the train distributions for one test week in June is given in Table 1. Each day we received between 399k and and 454k is-messages. Updating the distributions each minute with the newly arrived is-messages takes 140s to 162s for the whole day. So a server is less than 0.2% of the day busy with updates to the distributions. The average computation time per minute lies between 98ms an 113ms, the peak at 610ms to 790ms, still below one second.

The minor run-time fluctuations do not only depend on the number of messages or the different sizes of the timetable graphs (cf. Section 4.1), as we can see in the table. Additionally, the number of actual delays[2], the amount of time a train is delayed, the number of events dependent on the delayed events, the length of delayed trains, and the distribution over time of the delay messages[3] influence the computation time.

Distributions for connections. We calculated the distributions for 100,000 diverse connections obtained from answering real customer queries to our timetable information system MOTIS (see [5]). The average run-time per connection is 0.652ms. The minimum and maximum run-times are 0.362 ms and 0.916 ms, respectively.

4.3 Evaluation

4.3.1 Test Connections

We evaluated our model by periodically checking real connections. To do so, we queried MOTIS with a set of real queries combined with the top 100 relations in Germany[4], 8948

[2] some messages only state that a train is on time
[3] delays for earlier events potentially influence more distributions than delays for later events
[4] Most highly requested source-destination pairs as provided by Deutsche Bahn AG

relations in total.

Over 3 days we tracked 223,873 connections with an average of 1.5 transfers and an average duration of 231 minutes. We used MOTIS to check each connection and recompute the distributions and *rel*-ratings according to the up to then known delays a) before departure, b) every 75 minutes while traveling, and c) after arrival. From this data we created a subset of 76,095 connections for which we could ensure that the connection checker used real-time information from is-messages for most of its events at transfer stations. Connections without transfers do not have a *rel*-rating worth investigating. Hence, we removed direct connections from the dataset, leaving 63,524 samples.

We compared the predicted *rel*-rating before departure (see Section 3.2.2) to the actual connection feasibility. For this, we used the MOTIS check connection feature to determine whether all transfers of a connection are indeed feasible.

4.3.2 Evaluating Connection Reliability

Figure 2 compares the predicted *rel*-rating with the actual outcome. We grouped the *rel*-ratings to intervals of 10% plus an extra bin for the interval (99,100] and plotted the connection check results for each of those bins (see Figure 2). The dark area in bin (a, b] represents the percentage of feasible connections which had a reliability rating $\in$ (a, b]. Analogously, the bright area in bin (a, b] represents the percentage of infeasible connections which had a reliability rating $\in$ (a, b]. The numbers of connections assigned to the bins are different, and the width of each bin represents the number of connections in it. There are more connections with a higher rating than connections with a lower rating in the data set.

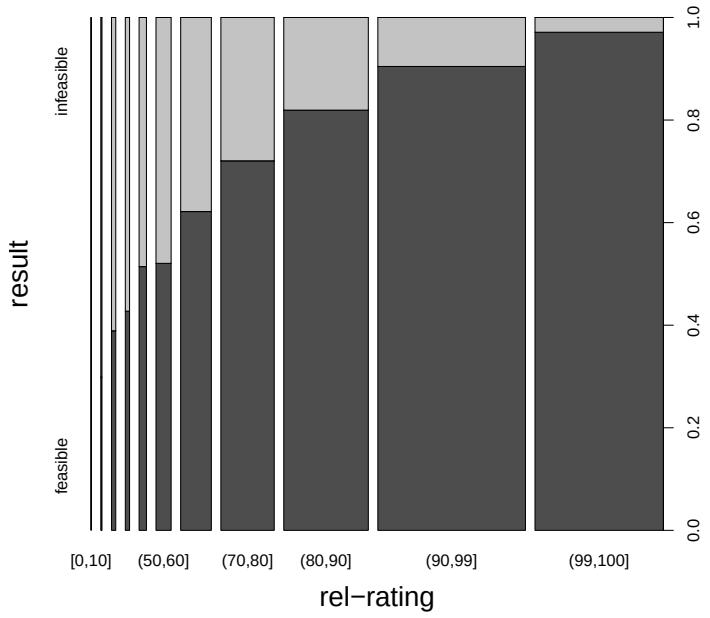

Figure 2 Predicted *rel*-rating versus actual outcome. Connections are grouped by their *rel*-ratings. Bar width represents number of connections in group. (light=connection infeasible, dark=connection feasible).

Table 2 Different *rel*-rating intervals with the share of all connections and the percentage of feasible ones of all connections in that interval.

rel-rating	Connections	
	% feasible	% of total
0−40	39.67	2.00
40−70	57.12	10.43
70−100	88.20	87.56

Table 3 Properties of the arrival distributions.

	Expected Delay	Breadth of Distribution
Min.	0.005	2.00
Median	0.715	9.00
Mean	1.476	13.29
Max.	21.823	61.00

Table 2 summarizes the data illustrated in the figure in larger intervals, corresponding to low, medium, and high reliability.

We found that of 49,071 connections with a *rel*-rating between 70% and 100%, the connection checker marked 6,568 or 11.8% as broken, while with 88.2% of the connections the passenger arrived at the target destination (Table 2). These connections account for 87.56% of the dataset. Connections with *rel*-ratings between 40% and 70% (10.4% of the dataset) were feasible in 57.1% of the cases. We see that for *rel*-ratings of more than 40%, the prediction was pretty accurate. In Figure 2 we can see that we are slightly optimistic for the intervals with a *rel*-rating higher than 70%. For *rel*-ratings lower than 40% the prediction was too conservative: fewer than predicted connections actually broke. This was the case for only 2% of tracked connections. The small sample size in that region might account for these results.

4.3.3 Analysis of Arrival Distributions

The evaluation of arrival distributions required us to ensure that the last arrival event of the connection was backed by an is-message. Also, only feasible connections are taken into account, further reducing the evaluation set to 31,620 connections.

4.3.3.1 Computed Distributions

Table 3 shows the expected values (interpreted as delays in minutes) and the breadth of the distributions. We define the breadth of a distribution as the minimal interval covering all non-zero probability values. A small average breadth distribution limits the necessary computation steps for estimating the individual arrival distributions. Furthermore, we see that the expected value for delays averaged over all connections is small but higher than 1, which is consistent to what we expect from the observed data.

4.3.3.2 Better Input Distributions

The analysis of our distributions and reliability ratings reveals room for improvement. We are sure that better input distributions will increase the quality of our results.

The travel time distributions play a crucial role in the quality of the arrival distributions. Presently, we generate synthetic travel time distributions which depend on the possible delays of departure events and scheduled travel times, in a preprocessing step. Our distributions already incorporate the potential of trains to catch up delays on driving sections as well as to get more delayed. The latter case could occur e.g. when delayed trains have to let

other trains to overtake. In the future, we will use travel time distributions provided by our cooperation partner, German Railways. These distributions are learned from months of real delay data.

Currently, we only consider the feeders for the first departure event of each train. In case the train has no feeders at the first stop or they arrive early enough, the probability that it departs on schedule equals 1. However, the departure can be delayed because of other factors like malfunctions, availability of the rails and trains, organizational issues, etc. We will receive *starting distributions* from German Railways for the first departures of the trains respecting these operational reasons. A convolution of our calculated departure distributions with these start distributions at the first departures would lead to more realistic results.

5 Conclusions and Future Work

5.1 Conclusion

We have presented a probabilistic approach for estimating the reliability of train connections. Several experiments on real customer queries and real timetables for all trains in Germany showed good results.

Initial propagation can be precomputed off-line in at most 75s. Updating with real-time information occupies the server less than 0.2% of the day. To determine the distribution for one connection takes less than 1ms.

We have shown that the predicted *rel*-ratings are valid approximations of the relative frequency of feasible connections. This could be verified by using real-time information for connection-checking. Only for the rare cases of very low *rel*-ratings, our predictions are too pessimistic. They are slightly optimistic for highly reliable connections and pretty accurate for the remaining ones.

5.2 Future Work

Use of more realistic distributions. We will integrate more realistic travel time distributions and starting distributions from German Railways to improve the quality of our predictions. In a later step, we plan to learn travel time distributions from is-messages and real timetable data regarding influential factors such as travel time, delay at departure, train category, stations on the route, weekday and daytime, and existing dependencies between trains.

Investigation of the independence assumptions. As mentioned in Section 3.1, the independence assumption is not always fulfilled. This implies that a departure distribution is not calculated correctly if there is a dependency between the arriving feeders. An analysis of the effect on the computed distributions is not trivial. We plan to further investigate this aspect with the use of our real timetable and delay data.

Comparison with a non-probabilistic approach. Once more realistic travel time distributions are used, it will be interesting to compare our model with other approaches. We plan a comparison with a non-probabilistic model which rates reliability of connections by analyzing the buffer times at the transfers in the connection.

Improved search for reliable connections. In this paper, we have shown that applying probability distributions is an effective approach for measuring the reliability of connections reasonably. We intend to integrate this probabilistic approach into our timetable information

system MOTIS [5] in order to provide searching for reliable connections. The first idea is to integrate the *rel*-rating as a new criterion in the multi-criteria search. A more complex approach is to find not only one reliable connection but a *connection graph* containing a reference connection and further alternative connections. The idea is to calculate the distributions not only on the basis of a single connection but considering several possible connections. The arrival distribution will then be composed of the distributions of the reference connection and all of its alternatives. Such a connection graph provides highest reliability for reaching the target station, and allows to reroute the passenger to an alternative connection if the *rel*-rating of the reference connection decreases.

Acknowledgments

Two of the authors were supported by German Railways Deutsche Bahn AG (RIS) through research contracts. We wish to thank Matthias Müller-Hannemann and his group at MLU Halle-Wittenberg and Christoph Blendinger from RIS for fruitful discussions.

References

1 Annabell Berger, Andreas Gebhardt, Matthias Müller-Hannemann, and Martin Ostrowski. Stochastic delay prediction in large train networks. In Alberto Caprara and Spyros C. Kontogiannis, editors, *ATMOS*, volume 20 of *OASICS*, pages 100–111. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Germany, 2011.

2 Malachy Carey and Andrzej Kwieciński. Stochastic approximation to the effects of headways on knock-on delays of trains. *Transportation Research Part B: Methodological*, 28(4):251 – 267, 1994.

3 Quan Lu, Maged Dessouky, and Robert C. Leachman. Modeling train movements through complex rail networks. *ACM Trans. Model. Comput. Simul.*, 14(1):48–75, January 2004.

4 L. E. Meester and S. Muns. Stochastic delay propagation in railway networks and phase-type distributions. *Transportation Research Part B*, 41:218–230, 2007.

5 Matthias Müller-Hannemann and Mathias Schnee. Finding all attractive train connections by multi-criteria pareto search. In Frank Geraets, Leo G. Kroon, Anita Schöbel, Dorothea Wagner, and Christos D. Zaroliagis, editors, *ATMOS*, volume 4359 of *Lecture Notes in Computer Science*, pages 246–263. Springer, 2004.

6 Matthias Müller-Hannemann and Mathias Schnee. Efficient timetable information in the presence of delays. In Ravindra K. Ahuja, Rolf H. Möhring, and Christos D. Zaroliagis, editors, *Robust and Online Large-Scale Optimization*, volume 5868 of *Lecture Notes in Computer Science*, pages 249–272. Springer, 2009.

7 Pavankumar Murali, Maged Dessouky, Fernando Ordóñez, and Kurt Palmer. A delay estimation technique for single and double-track railroads. *Transportation Research Part E: Logistics and Transportation Review*, 46(4):483 – 495, 2010. Selected papers from the Second National Urban Freight Conference, Long Beach, California, December 2007.

8 J. Yuan. *Stochastic modeling of train delays and delay propagation in stations*. PhD thesis, Technische Universiteit Delft, The Netherlands, 2006.